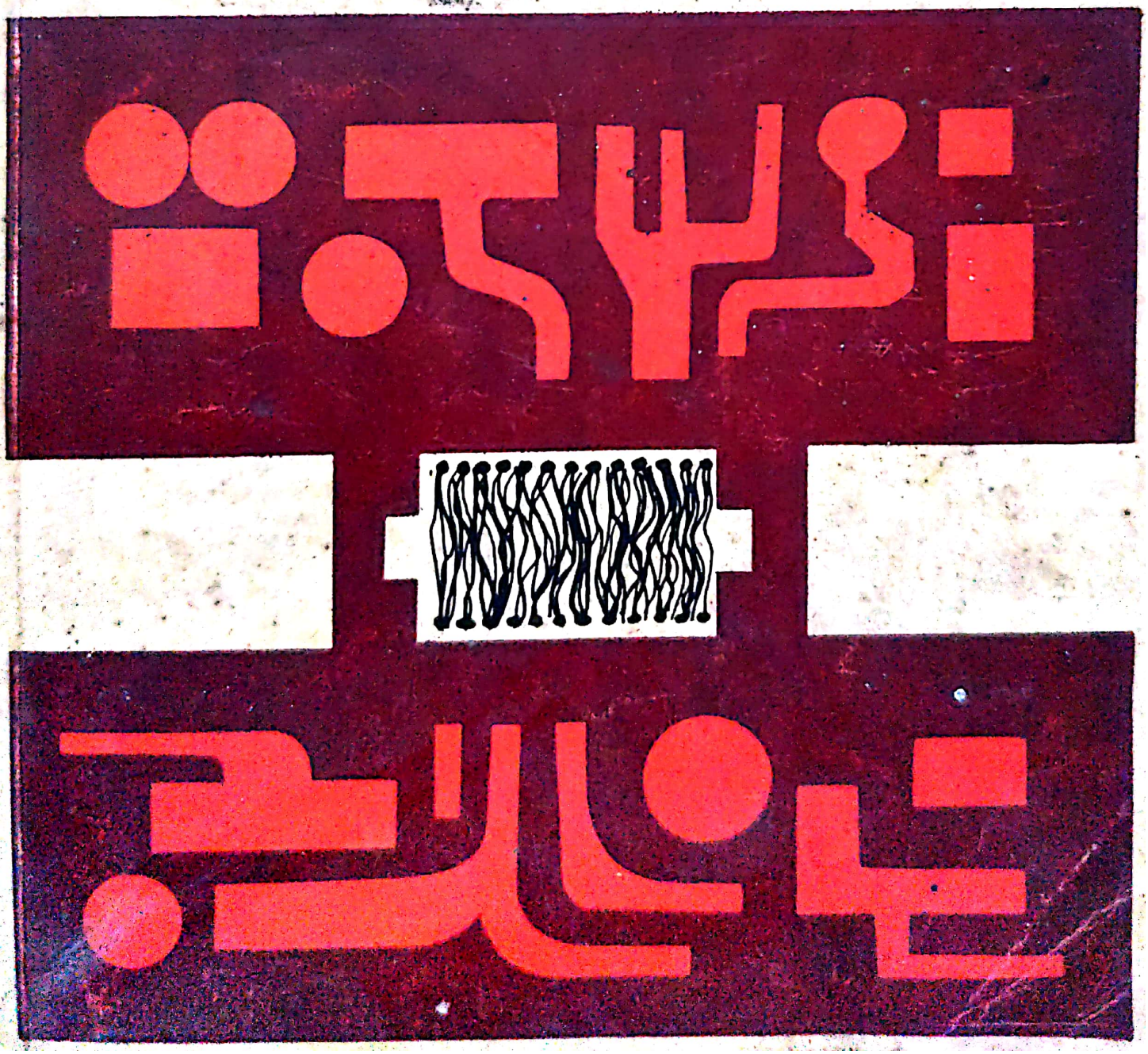
N. REUS

TRATAREA SIMBOLICĂ A SCHEMELOR DE COMUTAȚIE



Scanned with OKEN Scanner

Dr. ing. N. REUS

TRATAREA SIMBOLICĂ A SCHEMELOR DE COMUTAȚIE

EDITURA ACADEMIEI REPUBLICII SOCIALISTE ROMÂNIA
1971

Scanned with OKEN Scanner

TABLA DE MATERII

Introducere 7
Cap. 1. **Marcarea schemelor de comutație.** 9
Cap. 2. **Proprietățile raportului de stare** 17
Cap. 3. **Proprietățile simbolului de marcare** 26
3.1. Disjuncția logică 26
3.2. Conjuncția logică 32
3.3. Negația logică 34
3.4. Permutabilitatea indicilor. 42
3.5. Scoaterea în factor 46
Cap. 4. **Graful asociat unui simbol de marcare** 48
Cap. 5. **Clasificarea funcțiilor logice** 57
Cap. 6. **Metodă simbolică de minimizare a expresiilor booleene** . . 64
Cap. 7. **Metodă simbolică de analiză a schemelor de comutație** . . 82
Cap. 8. **Analiza schemei de automatizare a unei carde de bumbac** . 99
Cap. 9. **Transformarea schemelor cu contacte** 110
Cap. 10. **Minimizarea globală a expresiilor booleene** 119
Cap. 11. **Metodă simbolică de sinteză a schemelor de comutație.** . . 127
11.1. Sinteza schemelor de comandă automată a motoarelor electrice 127
11.2. Sinteza schemelor de comutație pentru acționări electrice cu cinematică programată 163
11.3. Sinteza unor scheme de comutație folosite în calculatoarele electronice și în automatizări industriale 207
11.4. Proiectarea logică a schemei de automatizare a unui laminor de bumbac. 240
11.5. Proiectarea logică a unei mașini de găurit automate (automat secvențial asincron). 252
Bibliografie 263
Symbolic treatment of commutation schemes (Summary) 269

Scanned with OKEN Scanner

INTRODUCERE

În etapa actuală de dezvoltare a tehnicii asistăm la o creștere impetuoasă a complexității sistemelor logice. Această creștere este determinată de extinderea pe scară din ce în ce mai largă a automatizării proceselor de producție, care devin din ce în ce mai complicate.

Sfera de utilizare a schemelor de comutație îmbrățișează domeniul acționărilor electrice, domeniul automatizărilor, telemecanicii, al conducerii programate a mașinilor unelte, telefonia automată, optimizatoarele numerice, calculatoarele numerice etc.

Complexitatea crescîndă a schemelor de comutație a impus efectuarea a numeroase cercetări privind găsirea unor metode de analiză și sinteză a acestor sisteme. În sfera acestor preocupări se încadrează și prezenta lucrare.

Teoria schemelor de comutație este de dată relativ recentă și se bazează pe algebra Boole, creată în 1847. Abia în 1910 Ehrenfest atrage atenția asupra posibilității de a aplica algebra Boole pentru studiul schemelor de comutație, iar în 1930 s-au lansat primele idei asupra folosirii algebrei lui Boole pentru studiul sistemelor logice, pe atunci realizate mai ales cu relee.

În 1938 americanul Claude Schannon a publicat un studiu în care se expunea aplicarea algebrei logicii la schemele cu contacte și relee. Acest studiu a stat la baza cercetărilor americane în acest domeniu.

În U.R.S.S., tot prin 1938, se pun bazele aplicării algebrei logicii la studiul schemelor cu contacte prin lucrările lui V. I. Șestakov și M. A. Gavrilov.

Printre lucrările mai importante în acest domeniu apărute în străinătate, mai amintim pe cele ale lui S. H. Caldwell, P. Naslin, J. Kuntzmann, S. Huffman, P. Tison, J. Florine etc.

Cercetările școlii românești în acest domeniu, inițiate de acad. Gr. C. Moisil, ocupă unul dintre primele locuri pe plan mondial. Aceste lucrări sînt citate în majoritatea lucrărilor de specialitate din acest domeniu.

Problemele de analiză și sinteză a schemelor de comutație fac obiectul a numeroase lucrări publicate de cercetători români (vezi bibliografia anexată). Aplicarea algebrei logicii pentru cazul particular al schemelor de comutație folosite în acționările electrice a făcut obiectul unor lucrări publicate de prof. dr. doc. N. V. Boțan

Scanned with OKEN Scanner

și colaboratorii săi de la Institutul politehnic din Iași. În aceste lucrări se arată posibilitatea aplicării algebrei logicii la analiza și la sinteza schemelor de acționare electrică.

Alături de metodele de analiză și sinteză a schemelor de comutație, bazate pe folosirea directă a formulei de interpolare a lui Lagrange, astăzi capătă răspîndire metode noi, care se grupează în metode algebrice, grafice și matriceale. Ele sînt tratate destul de amplu în literatura apărută în țara noastră. Analiza critică a acestor metode evidențiază faptul că încă nici una dintre ele nu satisface pe deplin cerințele practicii inginerești.

Prezenta lucrare își propune scopul de a oferi un nou punct de vedere în tratarea schemelor de comutație. Metoda simbolică de tratare a schemelor de comutație, care face obiectul acestei lucrări, se încadrează în efortul general de soluționare a problemelor inginerești de analiză și sinteză a schemelor de comutație. Aplicabilitatea ei se demonstrează prin problemele care au căpătat o soluție prin aplicarea metodei simbolice.

Scanned with OKEN Scanner

Capitolul 1 MARCAREA SCHEMELOR DE COMUTAȚIE

Schemele de comutaţie utilizate în sistemele automate moderne folosesc elemente cu funcţionare discretă, cum sînt diferitele tipuri de relee şi contactoare electromagnetice sau elemente de comutaţie statică, adică dispozitive electronice sau magnetice. Variabilele de intrare şi de ieşire ale acestor elemente sau ale schemei de comutaţie care le conţine pot lua numai valori discrete.

Dacă variabilele unei scheme de comutaţie pot avea numai valorile discrete 0 şi 1, ele se numesc binare. Valoarea binară 0 poate fi asociată absenţei semnalului de intrare sau absenţei acţiunii de la ieşire, iar valoarea binară 1 se poate asocia cu prezenţa semnalului de intrare sau cu prezenţa acţiunii la ieşire.

În general, o schemă de comutaţie are mai multe intrări şi ieşiri. Ea este deci caracterizată prin mai multe variabile binare de intrare şi de ieşire.

În grupul variabilelor binare de intrare sînt cuprinse variabilele primare şi cele secundare. Variabilele primare, iniţiatoare de comenzi, sînt independente faţă de cele de ieşire. Ele depind de acţiuni exterioare schemei de comutaţie, cum ar fi acţionarea diferitelor butoane sau comutatoare prin care se dau diferite comenzi.

Variabilele secundare, provenite de pe legăturile de reacţie ale schemei, depind de variabilele de ieşire ale acesteia.

Variabilele de ieşire ale unei scheme de comutaţie exprimă starea bobinelor de excitaţie ale releelor sau contactoarelor, prezenţa sau absenţa semnalelor electrice la ieşirea elementelor de comutaţie statică, prezenţa sau absenţa curentului în circuitul de excitaţie sau al indusului unui electromotor, starea becurilor semnalizatoare etc. Variabilele de ieşire pot avea sau nu ataşate variabile secundare.

În cazul schemelor de comutaţie dinamică, adică a schemelor de comutaţie care folosesc relee şi contactoare, variabilele de ieşire exprimă starea bobinelor de excitaţie ale acestora, iar variabilele

Scanned with OKEN Scanner

secundare atașate lor exprimă starea contactelor controlate de aceste relee sau contactoare.

În cazul schemelor de comutație statică, adică al schemelor care folosesc elemente de comutație fără contacte, cum sînt dispozitivele electronice sau magnetice, variabilele de ieșire exprimă prezența sau absența semnalelor electrice la ieșirea acestor elemente, iar variabilele secundare atașate lor exprimă existența sau absența legăturilor de reacție, adică a conexiunilor care leagă ieșirea acestor elemente cu intrările schemei de comutație.

Schemele de comutație care nu folosesc decît variabile primare, sînt scheme combinaționale. În acest caz, valorile binare ale variabilelor de ieșire ale schemei de comutație sînt determinate în întregime numai de combinația valorilor binare momentane ale variabilelor de intrare.

Introducerea unor legături de reacție în schemă, caracterizate prin variabile secundare atașate unor variabile de ieșire, transformă schema de comutație într-o schemă secvențială. În acest caz, valorile binare ale variabilelor de ieșire ale schemei de comutație depind și de timp. Această dependență se explică prin intervenția variabilelor secundare în etapa imediat următoare care succedă modificării ieșirilor de către variabilele de intrare.

Schemele secvențiale de comutație se caracterizează deci printr-o evoluție în timp. Dacă această evoluție este necontrolată din exterior prin variabile primare, schemele se numesc secvențiale asincrone. Dacă însă evoluția în timp a schemei secvențiale este controlată din exterior prin semnale de „tact" furnizate de un „generator de tacte", schema secvențială se numește sincronă.

În cazul schemelor secvențiale sincrone, aplicarea variabilelor secundare este controlată prin niște porți, care permit trecerea semnalelor de reacție numai simultan cu semnalele de „tact".

Starea unei scheme de comutație la un moment dat este caracterizată prin valorile binare ale variabilelor de intrare și de ieșire în acel noment. Dacă grupul variabilelor primare și al celor secundare se scrie într-o anumită ordine, atunci șirul valorilor binare ale lor poate caracteriza o stare a schemei de comutație.

Pentru o scriere mai concentrată, numărul binar, reprezentat de șirul valorilor binare ale variabilelor de intrare, se exprimă în codul octal, numindu-se „număr de stare". Se preferă codul octal, avîndu-se în vedere ușurința convertirii binar-octale și ușurința punerii în evidență a adiacențelor dintre numerele exprimate în codul octal.

Scanned with OKEN Scanner

Pentru a caracteriza complet o stare a schemei de comutație este necesară și indicarea valorilor binare ale variabilelor de ieșire ale schemei, care corespund numărului de stare considerat.

Starea unei scheme de comutație se poate reprezenta deci prin următorul simbol, pe care-l numim „raport de stare" :

$$\frac{n}{X_1X_2\ldots X_n}, \tag{1.1}$$

unde n este numărul de stare, iar $X_1X_2\ldots X_n$ șirul valorilor binare ale variabilelor de ieșire ale schemei de comutație.

Raportul de stare definit astfel se deosebește formal și structural de simboluri similare întîlnite în literatura de specialitate ([25] p. 57 și p. 155), [1], [23]).

Totalitatea stărilor posibile ale unei scheme de comutație caracterizează proprietățile funcționale ale acesteia. Alături de acestea, în cazul schemelor de comutație secvențiale, trebuie precizate și toate posibilitățile de evoluție în timp a stărilor schemei, adică secvențele de lucru și stările de blocare ale schemei. De asemenea, în cazul schemelor de comutație secvențiale asincrone, trebuie precizate și temporizările programate care se manifestă în evoluția în timp a schemei.

Exceptînd deocamdată aceste precizări, șirul valorilor binare ale unei variabile de ieșire, care corespund tuturor numerelor de stare posibile considerate în ordinea lor crescătoare, poate caracteriza proprietățile funcționale ale schemei referitoare la variabila de ieșire considerată.

Pentru o scriere mai concentrată, numărul binar, reprezentat de șirul valorilor binare ale variabilei de ieșire considerate, se exprimă în codul octal, numindu-se „număr de ordine".

Numărul de ordine definit astfel reprezintă expresia octală a „imaginii" unei funcții logice. Noțiunea de imagine a unei funcții logice este definită în [83 p. 14] prin succesiunea valorilor binare ale variabilei de ieșire pentru toate combinațiile posibile ale variabilelor de intrare, considerate în ordinea crescătoare a numerelor de stare.

O schemă de comutație cu mai multe variabile de ieșire poate fi deci marcată prin următorul simbol, denumit „simbol de marcare" :

$$F(X_1, X_2, \ldots, X_m) = D^{x_1x_2\ldots x_n}_{N_1, N_2\ldots N_m}, \tag{1.2}$$

unde N_k, este numărul de ordine referitor la variabila de ieşire X_k, iar $x_1x_2 \ldots x_n$, şirul ordonat al variabilelor de intrare. Ordinea de scriere a variabilelor de intrare corespunde ordinei folosite pentru determinarea numerelor de stare.

Simbolul de marcare (1.2) este scris în forma neoperativă. Pentru a putea opera cu el este convenabilă exprimarea lui într-o formă operativă. În acest scop, numerele de ordine N_k se înlocuiesc cu grupul numerelor de stare care corespund valorilor binare 1 ale variabilei de ieşire X_k, subînţelegîndu-se că restul numerelor de stare posibile corespund valorilor binare 0 ale acesteia.

În cazul sintezei unor scheme de comutaţie, pentru anumite stări ale acesteia nu sînt precizate valorile binare ale unor variabile de ieşire. Stările corespunzătoare ale schemei de comutaţie sînt deci indiferente. Ele corespund fie unor stări ce nu se pot ivi practic în funcţionarea schemei, fie unor stări ce nu influenţează îndeplinirea funcţiilor logice cerute schemei de comutaţie. Pentru o schemă de comutaţie cunoscută, toate numerele de stare sînt precis definite, neexistînd deci numere de stare indiferente.

Specificarea numerelor de stare indiferente este foarte utilă în procesul de sinteză al schemei, cînd schema de comutaţie nu este cunoscută. Această specificare se face printr-un asterisc (*), scris ca indice al acestor numere.

Dacă stările indiferente sînt mai numeroase decît cele care corespund valorilor binare 0 ale variabilei de ieşire, se preferă indicarea acestora în forma operativă a simbolului de marcare. Precizarea lor se face prin indicele zero (^{0}), ataşat acestor numere, sau specificînd separat numerele de stare care corespund valorii binare 1 şi valorii binare 0 a variabilei de ieşire.

Pentru o singură variabilă de ieşire, forma operativă a simbolului de marcare se va scrie

$$F(X) = D^{x_1x_2 \ldots x_n}_{(n_1, n_2 \ldots n_j^*, \ldots n_k)}. \tag{1.3}$$

Numerele de stare n_k din indicele inferior al simbolului de marcare se scriu între paranteze pentru a le distinge de numerele de ordine ale schemei de comutaţie.

În cazul mai multor variabile de ieşire, indicele inferior va cuprinde între paranteze, separat pentru fiecare variabilă, numerele de stare aferente acesteia, adică vom scrie

$$F(X_1, X_2 \ldots X_m) = D^{x_1x_2 \ldots x_n}_{(N_1), (N_2), \ldots (N_m)}. \tag{1.4}$$

Simbolul de marcare (1.3) sau (1.4) cuprinde în formă concentrată toate informaţiile care se dau obişnuit sub forma „tabelei de adevăr".

O tabelă de adevăr are n coloane, care corespund celor n variabile de intrare, şi 2^n linii, care corespund tuturor combinaţiilor posibile ale valorilor binare ale acestor variabile. De asemenea el cuprinde cîte o coloană pentru fiecare variabilă de ieşire, unde se înscriu valorile binare ale acestei variabile în corespondenţă cu diferitele combinaţii de intrare.

Raportul de stare (1.1) poate fi dedus din cel de marcare şi cu ajutorul lui se pot deduce secvenţele de lucru ale unei scheme de comutaţie.

Rezultă deci că simbolul de marcare (1.3) poate caracteriza complet proprietăţile funcţionale ale unei scheme de comutaţie, exceptînd temporizările programate care se manifestă în funcţionarea ei. Acest simbol, după cum se va arăta în cele ce urmează, poate fi tratat ca operator de calcul, spre deosebire de alte simboluri ([72] p. 101), ([25] p. 38), ([83] p. 15), care se folosesc numai în scopul unei simplificări a scrierii. El se dovedeşte foarte util în sinteza schemelor de comutaţie, folosindu-se în asociaţie cu raportul de stare definit mai sus.

Exemplul 1.1. Să se deducă simbolul de marcare şi raportul de stare pentru schema de comutaţie definită de tabela de adevăr 1.1. Pentru forma neoperativă a simbolului de marcare se vor considera stările indiferente (*) ca avînd valoarea binară 0.

Tabela de adevăr pentru acest exemplu cuprinde numai coloanele $x_0x_1x_2x_3$ pentru variabilele de intrare ale schemei şi coloanele X, Y şi Z pentru cele de ieşire.

Valorile binare indiferente ale variabilelor de ieşire sînt notate în această tabelă prin semnul (*). Ordinea liniilor corespunde sistemului de numeraţie binară, adică ordinii crescătoare a numerelor de stare n_k. Aceste numere de stare se înscriu într-o coloană ataşată tabelei de adevăr 1.1.

Folosind tabela 1.1, se scriu în ordinea crescătoare a numerelor de stare biţii reprezentativi ai celor trei variabile de ieşire, separîndu-se prin puncte grupe de cîte trei biţi, începînd cu ultimul bit. Se obţine astfel

$$(N_X)_2 = 0.110.000.101.100.111,$$
$$(N_Y)_2 = 0.011.001.100.101.001,$$
$$(N_Z)_2 = 0.101.001.101.101.011.$$

Expresia octală a acestor numere binare se obţine prin convertirea separată a grupelor de cîte trei biţi din codul binar în cel octal. Se obţine astfel

$$N_X = 60.547, \quad N_Y = 31.451 \quad \text{şi} \quad N_Z = 51.553.$$

Tabela 1.1.

Numărul de biţi „1"	Numărul de stare n_k	x_0	x_1	x_2	x_3	X	Y	Z
0	0	0	0	0	0	0	0	0
1	1	0	0	0	1	1	0	1
1	2	0	0	1	0	1	1	0
2	3	0	0	1	1	0	1	1
1	4	0	1	0	0	0	0	0
2	5	0	1	0	1	*	*	*
2	6	0	1	1	0	0	1	1
3	7	0	1	1	1	1	1	1
1	10	1	0	0	0	0	*	*
2	11	1	0	0	1	1	0	1
2	12	1	0	1	0	1	1	1
3	13	1	0	1	1	0	0	0
2	14	1	1	0	0	0	1	1
3	15	1	1	0	1	1	0	0
3	16	1	1	1	0	1	*	1
4	17	1	1	1	1	1	1	1

Forma neoperativă a simbolului de marcare se va scrie deci astfel

$$F(X, Y, Z) = D^{x_0 x_1 x_2 x_3}_{60.547,\ 31.451,\ 51.553.}$$

Forma operativă a simbolului de marcare se obţine direct din tabela de adevăr, folosind în cazul nostru coloana a doua a aces-

tuia în care sînt trecute numerele de stare. Această formă se poate exprima în cazul nostru fie evidențiind numerele de stare indiferente printr-un asterisc, fie pe cele care corespund valorii binare 0 a variabilei de ieșire printr-un indice (^{0}), fie evidențiind în paranteze separate numerele de stare care corespund valorilor binare 1 și 0 ale variabilei de ieșire.

Se poate deci scrie

$$F(X) = D^{x_0x_1x_2x_3}_{(1,\ 2,\ 5^*,\ 7,\ 11,\ 12,\ 15,\ 16,\ 17)} = D^{x_0x_1x_2x_3}_{(1,\ 2,\ 7,\ 11,\ 12,\ 15,\ 16,\ 17,\ 0^0,\ 3^0,\ 4^0,\ 6^0,\ 10^0,\ 13^0,\ 14^0)} =$$

$$= D^{x_0x_1x_2x_3}_{\substack{1:(1,\ 2,\ 7,\ 11,\ 12,\ 15,\ 16,\ 17),\\ 0:(0,\ 3,\ 4,\ 6,\ 10,\ 13,\ 14)}}$$

$$F(Y) = D^{x_0x_1x_2x_3}_{(2,\ 3,\ 5^*,\ 6,\ 7,\ 10^*,\ 12,\ 14,\ 16^*,\ 17)} = D^{x_0x_1x_2x_3}_{(2,\ 3,\ 6,\ 7,\ 12,\ 14,\ 17,\ 0^0,\ 1^0,\ 4^0,\ 11^0,\ 13^0,\ 15^0)} =$$

$$= D^{x_0x_1x_2x_3}_{\substack{1:(2,\ 3,\ 6,\ 7,\ 12,\ 14,\ 17)\\ 0:(0,\ 1{,}4,\ 11,\ 13,\ 15)}}$$

$$F(Z) = D^{x_0x_1x_2x_3}_{(1,\ 3,\ 5^*,\ 6,\ 7,\ 10^*,\ 11,\ 12,\ 14,\ 16,\ 17)} = D^{x_0x_1x_2x_3}_{(1,\ 3,\ 6,\ 7,\ 11,\ 12,\ 14,\ 16,\ 17,\ 0^0,\ 2^0,\ 4^0,\ 13^0,\ 15^0)} =$$

$$= D^{x_0x_1x_2x_3}_{\substack{1:(1,\ 3,\ 6,\ 7,\ 11,\ 12,\ 14,\ 16,\ 17)\\ 0:(0,\ 2,\ 4,\ 13,\ 15)}}$$

De regulă se preferă scrierea numerelor de stare în ordinea lor crescătoare și forma cea mai simplă de exprimare. În formă mai concentrată, simbolul de marcare pentru exemplul nostru se scrie

$$F(X, Y, Z) = D^{x_0x_1x_2x_3}_{\substack{(1,\ 2,\ 5^*,\ 7,\ 11,\ 12,\ 15,\ 16,\ 17),\\ (2,\ 3,\ 5^*,\ 6,\ 7,\ 10^*,\ 12,\ 14,\ 16^*,\ 17),\\ (1,\ 3,\ 5^*,\ 6,\ 7,\ 10^*,\ 11,\ 12,\ 14,\ 16,\ 17).}}$$

Scanned with OKEN Scanner

Raportul de stare pentru exemplul nostru se va scrie

$$\frac{x_0x_1x_2x_3}{XYZ} = \frac{n}{XYZ}.$$

Variabilele de ieşire neavînd ataşate şi variabilele scundare, schema de comutaţie va fi combinaţională.

Exemplul 1.2. Să se determine simbolul de marcare al blocului comparator a două numere binare formate din doi biţi, ştiind că ieşirea A are valoarea binară 1 pentru stările 4, 10, 11, 14, 15 şi 16 şi ieşirea B are valoarea binară 1 pentru stările 1,2,3,6,7, şi 13.

Forma operativă a simbolului de marcare rezultă imediat din enunţul problemei. Ea este

$$F(A, B) = D^{a_1a_0b_1b_0}_{(4,\ 10,\ 11,\ 14,\ 15,\ 16),\ (1,\ 2,\ 3,\ 6,\ 7,\ 13).}$$

unde a_1, a_0, b_1 şi b_0 sînt variabilele de intrare ale schemei care corespund biţilor reprezentativi ai celor două numere binare comparate.

Forma neoperativă a simbolului de marcare se obţine scriind mai întîi şirul celor $2^{2^2} = 16$ biţi reprezentativi ai numărului de ordine N al funcţiei logice. Pentru aceasta, numărînd în octal în sensul crescător de la 0 la 17, se completează cu 1 sau 0 valorile binare ale biţilor. Se formează apoi grupe de cîte trei biţi în scopul uşurării convertirii numărului N din codul binar în cel octal.

Se obţine astfel

$$(N_A)_2 = 0.000.100.011.001.110,$$

$$(N_B)_2 = 0.111.001.100.010.000.$$

Convertirea în codul octal ne conduce la

$$N_A = 4.316 \quad \text{şi} \quad N_B = 71.420.$$

Forma neoperativă a simbolului de macare va fi deci

$$F(A, B) = D^{a_1a_0b_1b_0}_{4.316,\ 71420.}$$

Capitolul 2 PROPRIETĂȚILE RAPORTULUI DE STARE

O stare a unei scheme de comutaţie poate fi stabilă sau instabilă.

În cazul cînd valorile binare ale variabilelor de ieşire coincid cu valorile binare ale variabilelor secundare, starea schemei este stabilă, iar în caz contrar starea acesteia este instabilă. Raportul de stare (1.1) ne indică valorile binare ale acestor variabile. Astfel, el poate evidenţia dacă o stare a schemei de comutaţie este stabilă sau instabilă.

Mai mult, în cazul unei stări instabile, acest raport permite determinarea stării următoare de tranziţie. Noua stare corespunde punerii în concordanţă a valorilor binare a variabilelor de ieşire şi secundare.

Aceasta se realizează prin modificarea valorii binare a variabilelor secundare astfel încît ele să fie egale valoric cu variabilele de ieşire ce le coresund.

Pentru a distinge grupul variabilelor secundare de cele primare, variabilele secundare se scriu între paranteze. De asemenea, în cazul cînd grupul variabilelor de ieşire conţine variabile de ieşire care nu au asociate şi variabile secundare, se scrie între paranteze grupul variabilelor de ieşire ce au asociate şi variabile secundare. În acest fel este uşurată identificarea stărilor stabile şi instabile ale schemei, identificare care se face prin compararea valorilor binare cuprinse între cele două paranteze de la numitorul şi numărătorul raportului de stare.

Ştiind simbolul de marcare al unei scheme de comutaţie şi discriminînd care dintre variabilele de intrare sînt primare şi care sînt secundare, se poate uşor structura raportul de stare al schemei în forma indicată mai sus. Pentru uşurarea comparării valorilor binare care vor fi înscrise în cele două paranteze de la numitorul şi numărătorul raportului de stare, se scriu în aceeaşi ordine variabilele de ieşire şi cele secundare ataşate lor. De asemenea, de regulă,

Scanned with OKEN Scanner

se notează prin litere mari variabilele de ieşite şi prin aceleaşi litere, dar mici, variabilele secundare ataşate lor.

Folosind simbolul de marcare în asociaţie cu raportul de stare, se pot stabili secvenţele de lucru şi stările de blocare care corespund schemei de comutaţie marcate prin aceste simboluri.

Secvenţele de lucru ale unei scheme de comutaţie se iniţiază prin modificarea valorilor binare iniţiale ale variabilelor primare. Trecerea dintr-o stare stabilă într-o altă stare nu este posibilă decît prin modificarea valorii binare a cel puţin uneia dintre variabilele primare. Aceste variabile pot fi deci privite ca iniţiatoare de comenzi. Din succesiunea stărilor de tranziţie se determină secvenţele de lucru ale schemei de comutaţie.

Modul de stabilire a acestor secvenţe prin folosirea informaţiilor cuprinse de simbolul de marcare şi raportul de stare va fi arătat prin exemplul (2.1).

Amintim că succesiunea stărilor stabile sau instabile prin care trece o schemă de comutaţie secvenţială poartă denumirea de secvenţă.

Pentru a caracteriza complet o stare a schemei de comutaţie este necesară cunoaşterea numărului de stare aferent ei şi a valorilor binare corespunzătoare variabilelor de ieşire. Aceste informaţii sînt cuprinse de raportul de stare. Adesea însă ne mulţumim cu o caracterizare incompletă, indicînd starea schemei numai prin numărul de stare. În scopul unei simplificări a expunerii, în cele ce urmează vom folosi adesea această formă de exprimare simplificată.

Exemplul 2.1. Se cere stabilirea secvenţelor de lucru ale schemei de comutaţie marcată prin simbolul

$$F(T, S, D) = D^{P\,tsd}_{63.146,\,21.643,\,51.523}.$$

Prin modul de notare al variabilelor, simbolul de marcare dat evidenţiază variabila primară P, variabilele secundare t, s şi d şi variabilele de ieşire T, S şi D.

Păstrînd ordinea variabilelor din indicele superior al simbolului de marcare, putem uşor structura următorul raport de stare:

$$\frac{P\,(tsd)}{TSD} = \frac{n}{TSD}.$$

Paranteza de la numitor nu mai este necesară în acest caz, întrucît toate variabilele de ieşire au asociate variabile secundare.

Se impune apoi găsirea formei operative a simbolului de marcare. În acest scop se convertesc în codul binar numerele de ordine ale schemei de comutaţie, obţinîndu-se

$$(N_T)_2 = 110.011.001.100.110,$$
$$(N_S)_2 = 10.001.110.100.011,$$
$$(N_D)_2 = 101.001.101.010.011.$$

Numerele de stare aferente fiecărei variabile de ieşire coincid ca număr cu numerele de ordine ale biţilor „1", făcînd numărătoarea în codul octal de la dreapta la stînga în sensul descrescător, adică începînd cu numărul de stare maxim.

Numărul de stare de valoare maximă pentru n variabile de intrare este dat de $(2^n-1)_8$. În cazul nostru, $n=4$ şi deci numărătoarea se va începe cu numărul $(16-1)_8 = 17$. Acest număr se poate găsi mai uşor prin convertirea în codul octal a numărului binar obţinut prin înlocuirea tuturor variabilelor de intrare cu biţi „1". Astfel, în cazul nostru avem $(1.111)_8 = 17$.

Procedînd în modul indicat mai sus, găsim următoarea formă operativă a simbolului de marcare :

$$F(T, S, D) = D^{Ptsd}_{\substack{(1,2,5,6,11,12,15,16),(2,6,7,10,12,16,17),\\(1,3,6,7,11,13,16,17).}}$$

Secvenţa de lucru a schemei, iniţiată prin variabila primară P, rezultă ca succesiune a stărilor

$$\frac{0}{000} \xrightarrow{P=1} \frac{01}{010} \rightarrow \frac{12}{110} \xrightarrow{\tau} \frac{16}{111} \rightarrow \frac{17}{011} \rightarrow \frac{13}{001} \rightarrow \frac{11}{101} \rightarrow$$

$$\xrightarrow{\tau} \frac{15}{100} \rightarrow \frac{14}{000} \rightarrow 10 \rightarrow \ldots$$

Valorile binare care corespund stării iniţiale a schemei de comutaţie, adică numărul de stare (0) rezultat din $P = t = s = d = 0$, rezultă

din simbolul de marcare. Acest simbol nu cuprinde numărul de stare (0), ceea ce semnifică $T = S = D = 0$.

Dacă $P = 1$, numărul de stare (0) trece în (10). Examinînd numerele de stare cuprinse de simbolul de marcare, constatăm că această stare corespunde următoarelor valori binare ale variabilelor de ieşire $T = D = 0$ şi $S = 1$. Rezultă deci că noua stare (10) este o stare instabilă, întrucît variabila secundară $s = 0$, deşi variabila de ieşire de care depinde, S, a luat valoarea binară 1. Practic, această stare ar putea corespunde situaţiei cînd bobina de excitaţie a unui releu S este străbătută de curent, dar contactele controlate de acest releu n-au avut încă timp să anclanşeze.

Evident, starea imediat următoarea corespunde punerii în concordanţă a valorilor binare ale variabilelor de ieşire din starea (10) cu variabilele secundare ataşate lor. Astfel, variabila secundară s devenind 1, schema trece într-o nouă stare, şi anume (12), care corespunde la $P = s = 1$ şi la $t = d = 0$, pentru care, după cum rezultă din simbolul de marcare, avem $T = S = 1$ şi $D = 0$. Această stare este de asemenea instabilă, ea trecînd, după timpul τ, în starea (16).

S-a considerat că variabila t este temporizată, căpătînd valoarea binară 1 cu întîrzierea τ. Prin modul de notare a acestei întîrzieri în schema secvenţei date mai sus, se precizează şi temporizările controlabile ce se manifestă în funcţionarea unei scheme secvenţiale asincrone. În acest fel se completează informaţiile asupra funcţionării schemei şi cu ultimele precizări necesare.

Procedînd în modul arătat mai sus, se stabilesc următoarele stări de tranziţie ale schemei, şi anume din starea (12) în stările (16), (17), (13), (11), (15), (14), (10), (12), (16),... Se constată astfel că schema secvenţei de lucru evidenţiază în cazul nostru particular o funcţionare ciclică. Schema se comportă deci ca un oscilator autonom atît timp, cît $P = 1$.

Dacă variabila primară P redevine 0 începînd cu starea (12), atunci secvenţa de lucru a schemei, dedusă la fel ca secvenţa anterioară, va fi

$$(12) \xrightarrow{P=0} \frac{2}{110} \xrightarrow{\tau} \frac{6}{111} \rightarrow \frac{7}{011} \rightarrow \frac{3}{001} \rightarrow \frac{1}{101} \xrightarrow{\tau}$$

$$\rightarrow \frac{5}{100} \rightarrow \frac{4}{000} \rightarrow (0).$$

Schema secvenţei de lucru evidenţiază în acest caz o funcţionare neciclică.

Exemplul 2.2. Se cere stabilirea secvenţelor de lucru ale schemei de comutaţie, marcată prin simbolul

$$F(S, X, Y) = D^{Pxy}_{(3,6,7),(4,6,7),(4,5,7)}.$$

Din simbolul de marcare dat rezultă raportul de stare

$$\frac{P(xy)}{S(XY)} = \frac{n}{SXY}.$$

Secvenţele de lucru ale schemei, iniţiate prin variabila primară P, sînt date de schema

$$\begin{array}{l} \nearrow \dfrac{5}{100} \xrightarrow{P=0} \dfrac{1}{100} \to (0), \\ \dfrac{0}{000} \xrightarrow{P=1} \dfrac{4}{011} \longrightarrow \dfrac{7}{111} \xrightarrow{P=0} \dfrac{3}{100} \begin{array}{l} \nearrow (1) \to (0), \\ \searrow (2) \to (0), \end{array} \\ \qquad\qquad \searrow \dfrac{6}{110} \xrightarrow{P=0} \dfrac{2}{000} \to (0). \end{array}$$

Remarcăm mai întîi că în starea (4) devin 1 simultan două variabile de ieşire care au ataşate variabile secundare, şi anume $X = Y = 1$. De la această stare sînt posibile trei tranziţii, în funcţie de care dintre variabilele secundare ia mai întîi valoarea binară 1.

Practic, în cazul schemelor cu relee se poate ivi situaţia cînd contactele releului Y anclanşează înaintea contactelor releului X, schema trecînd în starea stabilă (5), situaţia inversă cînd schema trece în starea (6) şi situaţia puţin probabilă cînd contactele ambelor relee anclanşează simultan, schema trecînd în starea stabilă (7). Rezultatul cursei semnalelor de ieşire depinde de hazard, fenomenul purtînd chiar denumirea de „hazard de curse". În cazul examinat mai sus, cursa se numeşte „critică", întrucît stările finale, stabile, diferă între ele.

Scanned with OKEN Scanner

În schema secvențelor redată mai sus mai remarcăm încă o situație, cînd are loc schimbarea simultană a valorii binare a două variabile de ieșire care au atașate variabile secundare. Astfel, dacă în starea (7) facem $P = 0$, are loc simultan $X = 0$ și $Y = 0$. Drept rezultat se va manifesta din nou fenomenul amintit mai sus. Întrucît însă starea finală (0) este aceeași indiferent de stările de tranziție prin care trece schema, cursa se numește „necritică".

În concluzie facem constatarea că prezența hazardului de curse este evidențiată de schema secvențelor de lucru ale unei scheme de comutație, în care stările schemei se exprimă cu ajutorul raportului de stare.

Acest tip de hazard apare întotdeauna cînd trecerea dintr-o stare instabilă într-o altă stare impune modificarea simultană a valorii binare a două sau a mai multor variabile secundare.

Exemplul 2.3. Se cere precizarea stărilor stabile ale schemei de comutație date prin simbolul

$$F(X, Y) = D^{abxy}_{(6,7,14,16,17),(2,3,6,7,10,11,12,13,15,17)}.$$

Plecînd de la aceste stări și folosind regula elaborată de Unger [97], să se precizeze dacă schema prezintă hazard esențial (de propagare sau hazard dinamic).

Din simbolul de marcare dat rezultă raportul de stare

$$\frac{ab(xy)}{XY} = \frac{n}{XY}.$$

Scriind raporturile de stare pentru toate cele $2^n = 16$ numere de stare posibile pentru $n = 4$ variabile de intrare, constatăm că identitatea valorilor binare a variabilelor de ieșire și a acelor secundare atașate lor este satisfăcută numai de stările (0), (4), (7), (11), (15), (16) și (17). Acestea vor fi stările stabile ale schemei de comutație. Restul celor $16 - 7 = 9$ stări ale schemei vor fi stări de tranziție instabile.

Dacă schema de comutație este electronică, se poate întîmpla ca timpii de propagare ai aceluiași semnal primar pe diferite trasee ale circuitului logic care elaborează semnalele secundare să fie dife-

riţi. Drept urmare a acestui fenomen, denumit „hazard de propagare", starea finală a unei scheme de comutaţie poate depinde de viteza de propagare a semnalelor primare.

Hazardul de propagare nu se manifestă în circuitele cu relee, în care timpii de propagare sînt întotdeauna neglijabili în raport cu timpii de basculare ai releelor. În schimb, în cazul acestor circuite se manifestă un alt tip de hazard, denumit „hazard dinamic". El este cauzat de vibrarea contactelor care face ca o variabilă primară să-şi schimbe în primul moment valoarea, apoi să revină la valoarea iniţială etc. şi abia după o serie de vibraţii de acest fel să capete definitiv valoarea finală impusă din exterior.

După cum se arată în lucrarea [72, p. 369] şi [102], Unger a stabilit o regulă (criteriu) de evidenţiere a hazardului de propagare, denumit de el şi „hazard esenţial". Noi considerăm că această regulă este valabilă şi pentru evidenţierea hazardului dinamic, avînd în vedere conţinutul ei.

Potrivit criteriului lui Unger, o schemă de comutaţie prezintă hazard de propagare dacă trei schimbări consecutive ale aceleiaşi variabile primare nu conduc schema la aceeaşi stare ca o schimbare unică a acestei variabile.

În practică este necesar să evidenţiem prezenţa hazardului de propagare sau dinamic numai pentru secvenţele de lucru impuse, adică pentru variaţiile variabilelor primare impuse de specificaţiile de lucru ale schemei.

În cazul exemplului nostru, vom verifica existenţa hazardului de propagare sau dinamic pentru toate secvenţele posibile care pot fi iniţiate prin schimbarea valorii binare a unei singure variabile binare, plecînd de la cele şapte stări stabile ale schemei date. Se obţin astfel secvenţele

$$1)\quad \frac{0}{00} \xrightarrow{a=1} \frac{10}{01} \to \frac{11}{01} \xrightarrow{a=0} \frac{1}{00} \to (0) \xrightarrow{a=1} (10),$$

$$2)\quad (0) \xrightarrow{b=1} \frac{4}{00} \xrightarrow{b=0} (0) \xrightarrow{b=1} (4),$$

$$3)\quad (4) \xrightarrow{a=1} \frac{14}{10} \to \frac{16}{10} \xrightarrow{a=0} \frac{6}{11} \to \frac{7}{11} \xrightarrow{a=1} \frac{17}{11} \xrightarrow{a=0} (7) \xrightarrow{a=1} (17),$$

4) $(7) \xrightarrow{b=0} \dfrac{3}{01} \to (1) \to (0) \xrightarrow{b=1} (4) \xrightarrow{b=0} (0),$

5) $(11) \xrightarrow{a=0} (1) \to (0) \xrightarrow{a=1} (10) \to (11) \xrightarrow{a=0} (1) \to (0),$

6) $(11) \xrightarrow{b=1} \dfrac{15}{01} \xrightarrow{b=0} (11) \xrightarrow{b=1} (15),$

7) $(15) \xrightarrow{a=0} \dfrac{5}{00} \to (4) \xrightarrow{a=1} (14) \to (16) \xrightarrow{a=0} (6) \to (7),$

8) $(15) \xrightarrow{b=0} (11) \xrightarrow{b=1} (15) \xrightarrow{b=0} (11),$

9) $(16) \xrightarrow{b=0} \dfrac{12}{01} \begin{matrix} \nearrow (10) \\ \searrow (13) \end{matrix},$

10) $(17) \xrightarrow{a=0} (7) \xrightarrow{a=1} (17) \xrightarrow{a=0} (7),$

11) $(17) \xrightarrow{b=0} \dfrac{13}{01} \to (11) \xrightarrow{b=1} (15) \xrightarrow{b=0} (11).$

Remarcăm prezenţa hazardului de propagare (în schema electronică) sau dinamic (în schema electronică sau cu relee) care apare în secvenţele 3) şi 7).

În cazul secvenţei 3), plecînd din starea (4) şi efectuînd o singură schimbare a variabilei primare ($a = 1$), schema ajunge în starea stabilă (16), iar după trei schimbări succesive ale acestei variabile se ajunge în starea stabilă (17).

În cazul secvenţei 7), plecînd din starea (15) şi efectuînd o singură schimbare a variabilei primare ($a = 0$), schema ajunge în starea stabilă (4), iar după trei schimbări succesive ale acestei

Scanned with OKEN Scanner

variabile se ajunge în starea (7). Secvența 9) prezintă hazard static.

Făcînd bilanțul proprietăților raportului de stare, putem afirma că el poate servi pentru descrierea stărilor unei scheme de comutație, precizînd care dintre acestea sînt stabile și care sînt tranzitorii, adică instabile. Cu ajutorul lui se pot determina posibilitățile de tranziție dintr-o stare în alta.

De asemenea, raportul de stare oferă posibilitatea ca în asociație cu simbolul de marcare să se stabilească secvențele de lucru ale unei scheme secvențiale de comutație. Aceste secvențe evidențiază prin rapoartele stărilor funcționarea ciclică sau neciclică a schemei, prezența hazardului de curse și tipul lui, a hazardului de propagare și a celui dinamic.

Scanned with OKEN Scanner

Capitolul 3 PROPRIETĂȚILE SIMBOLULUI DE MARCARE

În afară de faptul că simbolul de marcare exprimă în formă concentrată proprietăţi ale unei scheme de comutaţie, el poate fi tratat şi ca operator de calcul.

În cele ce urmează ne vom ocupa de acele proprietăţi de operator ale simbolului de marcare care se dovedesc utile în analiza şi sinteza schemelor de comutaţie. Astfel, alături de alte proprietăţi, simbolul de marcare permite aplicarea legilor algebrei booleene pentru cazul cînd locul variabilelor binare este luat de conjuncţii ale unor astfel de variabile.

3.1. DISJUNCȚIA LOGICĂ

Fie o schemă de comutaţie ale cărei m variabile de ieşire depind toate de cele n variabile de intrare. Aceste dependenţe sînt date obişnuit sub forma tabelei de adevăr. Ele definesc o anume funcţie logică, exprimată printr-o anumită formulă a algebrei booleene.

Folosirea formulei de interpolare a lui Lagrange permite ca direct din tabela de adevăr să obţinem două forme algebrice de exprimare a conţinutului acesteia. Prima formă a funcţiei logice, numită şi „formă canonică disjunctivă", se obţine prin scrierea disjuncţiei tuturor conjuncţiilor variabilelor de intrare, directe sau negate, ale căror combinaţii corespund valorii binare 1 a variabilei de ieşire :

$$F(X) = \bigcup_{p=1}^{j} \left(\prod_{k=1}^{n} x_k^{i_k} \right)_p . \tag{3.1}$$

Fiecare dintre cele n variabile de intrare care intră în conjuncţiile acestei forme canonice se scrie nenegată dacă valoarea binară

cu care intră în combinație este 1 și negată dacă această valoare binară este 0. Acest lucru se obține cu formula (3.1) dacă pentru $i_k = 1$ se ia $x_k^{i_k} = x_k$ și dacă pentru $i_k = 0$ se ia $x_k^{i_k} = \bar{x}_k$.

A doua formă a funcției logice, numită „formă canonică conjunctivă", se obține prin scrierea conjuncției tuturor disjuncțiilor variabilelor de intrare directe sau negate ale căror combinații corespund valorii binare 0 a variabilelor de ieșire:

$$F(X) = \prod_{p=1}^{j} \left(\bigcup_{k=1}^{n} x_k^{i_k} \right)_p . \tag{3.2}$$

Fiecare dintre cele n variabile de intrare care intră în disjuncțiile acestei forme canonice se scrie nenegată dacă valoarea binară cu care intră în combinație este 0 și negată dacă această valoare binară este 1. Acest lucru se obține cu formula (3.2), făcînd convenția inversă celei anterioare, adică pentru $i_k = 1$ luăm $x_k^{i_k} = \bar{x}_k$ și pentru $i_k = 0$ luăm $x_k^{i_k} = x_k$.

Rezultă deci că în cazul formei canonice conjunctive vor trebui considerate numerele de stare care se găsesc prin negarea logică a biților reprezentativi dați de tabela de adevăr, care reprezintă deci numerele de stare în codul binar invers. Aceste numere de stare sînt negatele logice ale celor ce intervin în forma canonică disjunctivă a funcției logice.

Facem convenția ca formula (3.1) să fie exprimată de simbolul de marcare, adică

$$\bigcup_{p=1}^{j} \left(\prod_{k=1}^{n} x_k^{i_k} \right)_p = D_{(n_1, n_2, \ldots n_j)}^{x_1 x_2 \ldots x_n} . \tag{3.3}$$

Forma canonică conjunctivă a unei funcții logice (3.2) se poate de asemenea reprezenta prin simbolul

$$\prod_{p=i}^{s} \left(\bigcup_{k=1}^{n} x_k^{i_k} \right)_p = C_{(\bar{n}_i, \bar{n}_h, \ldots \bar{n}_s)}^{x_1 x_2 \ldots x_n} . \tag{3.4}$$

Formulele (3.1)—(3.4) reprezentînd diferitele forme ale aceleiași funcții logice, vom mai putea scrie

$$D_{(n_1, n_2, \ldots n_j)}^{x_1 x_2 \ldots x_n} = C_{(\bar{n}_i, \bar{n}_h, \ldots \bar{n}_s)}^{x_1 x_2 \ldots x_n} . \tag{3.5}$$

Scanned with OKEN Scanner

Trebuie să remarcăm că formula (3.5) este valabilă numai dacă numerele de stare pentru simbolul D se determină pentru valorile binare 1 ale variabilei de ieşire, iar numerele de stare pentru simbolul C se determină pentru valorile binare 0 ale acestei variabile, adică mulţimile :

$$(n_1, n_2, \ldots, n_j) \quad \text{şi} \quad (n_i, n_h, \ldots, n_s), \quad \text{sînt disjuncte}$$

În aplicaţiile practice, pentru a obţine o tratare mai unitară a problemelor, vom folosi, de regulă, numai simbolul D.

Întotdeauna cînd nu vom specifica în cadrul acestei lucrări simbolul, vom subînţelege că este vorba de simbolul de marcare D.

Întrucît pentru cazul particular cînd $j = 1$ formula (3.3) se scrie

$$F_1(X) = \prod_{k=1}^{n} x_k^{i_k} = D_{(n_1)}^{x_1 x_2 \cdots x_n}, \tag{3.6}$$

convenţia de definiţie a simbolului de marcare se mai poate pune sub forma

$$F(X) = \bigcup_{p=1}^{j} D_{(n_p)}^{x_1 x_2 \cdots x_n}, \tag{3.7}$$

unde n_p reprezintă numerele de stare care corespund unor valori binare 1 sau indiferente ale variabilei de ieşire, iar $D_{(n_p)}^{x_1 x_2 \cdots x_n}$ conjuncţia variabilelor de intrare directe sau negate în corespondenţă cu specificaţia numărului de stare n_p. Potrivit acestei specificaţii, variabilele de intrare cu valoarea 1 în combinaţia respectivă apar în conjuncţie în forma directă (nenegată), iar cele cu valoarea 0 apar în conjuncţie sub forma negată.

Exemplul 3.1. Ne propunem ca, folosind simbolurile de marcare deduse în exemplul 1.1, să găsim cele două forme canonice ale funcţiei logice $F(X)$. Presupunem că schema de comutaţie care corespunde acestui simbol are precizate stările indiferente prin valoarea binară „0".

Scanned with OKEN Scanner

Avînd în vedere considerațiile de mai sus, putem scrie imediat forma canonică disjunctivă a funcției logice :

$$F(X) = D^{x_0x_1x_2x_3}_{(1,2,7,11,12,15,16,17)} =$$

$$= \bar{x}_0\bar{x}_1\bar{x}_2x_3 \cup \bar{x}_0\bar{x}_1x_2\bar{x}_3 \cup \bar{x}_0x_1x_2x_3 \cup x_0\bar{x}_1\bar{x}_2x_3 \cup$$

$$\cup x_0\bar{x}_1x_2\bar{x}_3 \cup x_0x_1\bar{x}_2x_3 \cup x_0x_1x_2\bar{x}_3 \cup x_0x_1x_2x_3.$$

Folosind convenția de definiție pentru forma canonică conjunctivă a funcției logice, obținem

$$F(X) = C^{x_0x_1x_2x_3}_{(\overline{0},\overline{3},\overline{4},\overline{5},\overline{6},\overline{10},\overline{13},\overline{14})} = C^{x_0x_1x_2x_3}_{(3,4,7,11,12,13,14,17)} =$$

$$= (\bar{x}_0 \cup \bar{x}_1 \cup x_2 \cup x_3)(\bar{x}_0 \cup x_1 \cup \bar{x}_2 \cup \bar{x}_3) \times$$

$$\times (\bar{x}_0 \cup x_1 \cup x_2 \cup x_3)(x_0 \cup \bar{x}_1 \cup \bar{x}_2 \cup x_3) \times$$

$$\times (x_0 \cup \bar{x}_1 \cup x_2 \cup \bar{x}_3)(x_0 \cup \bar{x}_1 \cup x_2 \cup x_3) \times$$

$$\times (x_0 \cup x_1 \cup \bar{x}_2 \cup \bar{x}_3)(x_0 \cup x_1 \cup x_2 \cup x_3).$$

Se poate arăta prin transformări că această formulă este identică ca valoare logică cu cea obținută pentru forma canonică disjunctivă.

Remarcăm că prin folosirea simbolului de marcare se poate deduce funcția logică la fel ca prin folosirea tabelei de adevăr.

Avînd în vedere convențiile simbolului de marcare și proprietatea de asociativitate a disjuncției logice, putem scrie

$$D^{x_1x_2\ldots x_n}_{(n_1, n_2, \ldots n_k)} = D^{x_1x_2\ldots x_n}_{(n_1, \ldots n_j)} \cup \ldots \cup D^{x_1x_2\ldots x_n}_{(n_i, \ldots, n_k)}. \quad (3.8)$$

Rezultă deci că un simbol de marcare se poate exprima prin disjuncția mai multor simboluri de marcare, conținînd fiecare unul sau mai multe dintre numerele de stare care caracterizează acest simbol. Unele dintre aceste simboluri se pot repeta și de asemenea se pot repeta unele numere de stare care intervin în aceste simboluri. Aceasta este o consecință a legii de idempotență din algebra booleeană, care în formă simbolică se scrie

$$D^{x_1x_2\ldots x_n}_{(n_i, \ldots n_i)} = D^{x_1x_2\ldots x_n}_{(n_i)}. \quad (3.9)$$

Scanned with OKEN Scanner

Din formulele (3.8) şi (3.9) mai rezultă că, dacă drept rezultat al disjuncţiei mai multor simboluri de marcare în simbolul de marcare rezultant apar numere de stare multiple (cu aceeaşi valoare), atunci în acest simbol rămîne cîte un singur reprezentant pentru fiecare grup comun.

În fine, legea absorbţiei din algebra booleeană se traduce în cazul utilizării simbolului de marcare prin următoarea formulă:

$$D_{(n_i)}^{x_1 x_2 \dots x_n} \cup D_{(n_i,\, n_k)}^{y_1 x_1 x_2 \dots x_n} \cup D_{(n_i,\, n_j)}^{y_2 x_1 x_2 \dots x_n} \cup \dots$$

$$\dots \cup D_{(n_i,\, n_f)}^{y_k x_1 x_2 \dots x_n} = D_{(n_i)}^{x_1 x_2 \dots x_n}. \tag{3.10}$$

Remarcăm că formula (3.10) nu este unica formă în care se traduce simbolic legea absorbţiei din algebra booleeană. Astfel se pot ivi cazuri cînd variabila absorbită y_k poate avea şi alt loc în şirul variabilelor din indicele superior al simbolului de marcare, acest indice poate conţine mai multe variabile de felul lui y_k şi de asemenea astfel de variabile pot reprezenta, la rîndul lor, funcţii logice oarecare. În aceste cazuri este însă posibil întotdeauna ca prin transformări (substituţii) să aducem simbolurile de marcare la formele cuprinse de formula (3.10).

Alte legi ale algebrei booleene referitoare numai la disjuncţie, cum ar fi legile lui 1 şi 0, nu evidenţiază proprietăţi deosebite în forma simbolică, cînd locul variabilelor binare este luat de funcţii logice exprimate prin simboluri de marcare.

În cazul cînd locul simbolului D este luat de simbolul C, formulele (3.6)— (3.10) se scriu

$$F_1(X) = \bigcup_{k=1}^{n} x_k^{i_k} = C_{(\overline{n}_1)}^{x_1 x_2 \dots x_n}, \tag{3.11}$$

$$F(X) = \prod_{p=i}^{s} C_{(\overline{n}_i)}^{x_1 x_2 \dots x_n}, \tag{3.12}$$

$$C_{(\overline{n}_i, \dots \overline{n}_s)}^{x_1 x_2 \dots x_n} = C_{(\overline{n}_i, \dots \overline{n}_j)}^{x_1 x_2 \dots x_n} \dots C_{(\overline{n}_h, \dots \overline{n}_s)}^{x_1 x_2 \dots x_n}, \tag{3.13}$$

$$C^{x_1 x_2 \ldots x_n}_{(\overline{n}_i, \ldots \overline{n}_i)} = C^{x_1 x_1 \ldots x_n}_{(\overline{n}_i)}, \tag{3.14}$$

$$C^{x_1 x_2 \ldots x_n}_{(\overline{n}_i)} \cdot C^{y_1 x_1 x_2 \ldots x_n}_{(\overline{n}_i, n_k)} \cdot C^{y_2 x_1 x_2 \ldots x_n}_{(\overline{n}_i, n_j)} \ldots$$

$$\ldots C^{y_k x_1 x_2 \ldots x_n}_{(\overline{n}_i, n_f)} = C^{x_1 x_2 \ldots x_n}_{(\overline{n}_i)}. \tag{3.15}$$

Exemplul 3.2. Să se verifice valabilitatea formulelor (3.7)—(3.10), pentru cazul unor funcții logice de două variabile binare.

Pentru a verifica valabilitatea formulei (3.7), să considerăm funcția logică definită prin simbolul de marcare

$$F(X) = D^{x_0 x_1}_{(0, 2)}.$$

Evident, vom avea

$$D^{x_0 x_1}_{(0, 2)} = \bar{x}_0 \bar{x}_1 \cup x_0 \bar{x}_1 = D^{x_0 x_1}_{(0)} \cup D^{x_0 x_1}_{(2)}.$$

Pentru a verifica valabilitatea formulelor (3.8) și (3.9), să considerăm funcția logică definită prin

$$F(Y) = D^{x_0 x_1}_{(0, 1, 3)} \cup D^{x_0 x_1}_{(0, 3)}.$$

Exprimînd separat fiecare simbol prin funcțiile logice pe care le definește și efectuînd reducerile posibile prin aplicarea legilor algebrei booleene, obținem

$$F(Y) = (\bar{x}_0 \bar{x}_1 \cup \bar{x}_0 x_1 \cup x_0 x_1) \cup (\bar{x}_0 \bar{x}_1 \cup x_0 x_1) =$$

$$= \bar{x}_0 \bar{x}_1 \cup \bar{x}_0 x_1 \cup x_0 x_1 = D^{x_0 x_1}_{(0, 1, 3)}.$$

Rezultatul obținut verifică formula (3.8) și el s-a obținut și prin aplicarea formulei (3.9).

Scanned with OKEN Scanner

În fine, pentru a verifica valabilitatea formulei (3.10), să considerăm funcţia logică definită prin

$$F(Z) = D_{(2)}^{x_0 x_1} \cup D_{(2,\,6)}^{y_0 x_0 x_1} \cup D_{(2)}^{y_1 x_0 x_1}.$$

Explicitînd termenii acestei expresii logice şi efectuînd reducerile posibile prin aplicarea legilor algebrei boleene, obţinem

$$F(Z) = x_0 \bar{x}_1 \cup \bar{y}_0 x_0 \bar{x}_1 \cup \bar{y}_1 x_0 \bar{x}_1 \cup y_0 x_0 \bar{x}_1 = x_0 \bar{x}_1 (1 \cup \bar{y}_0 \cup \bar{y}_1) =$$

$$= x_0 \bar{x}_1 = D_{(2)}^{x_0 x_1},$$

care verifică relaţia (3.10).

3.2. CONJUNCŢIA LOGICĂ

Să considerăm conjuncţiile logice care corespund la două numere de stare diferite. În acest caz, cel puţin o variabilă apare în forma directă într-una din conjuncţii şi în forma negată în cealaltă, aşţfel încît produsul lor logic (conjuncţia), ca urmare a legii contradicţiei, rezultă nul, adică

$$D_{(a)}^{x_1 x_2 \ldots x_n} \cdot D_{(b)}^{x_1 x_2 \ldots x_n} = 0, \tag{3.16}$$

dacă $a \neq b$.

Dacă $a = b$, ca urmare a legii idempotenţei, avem

$$D_{(a)}^{x_1 x_2 \ldots x_n} \cdot D_{(b)}^{x_1 x_2 \ldots x_n} = D_{(a)}^{x_1 x_2 \ldots x_n} = D_{(b)}^{x_1 x_2 \ldots x_n}. \tag{3.17}$$

Efectuînd conjuncţia logică a unor simboluri de marcare cu acelaşi indice superior şi conţinînd fiecare mai multe numere de stare, simbolul rezultant va conţine numai numerele de stare comune simbolurilor care intră în conjuncţie. Această concluzie este o consecinţă directă a relaţiilor (3.16) şi (3.17).

Scanned with OKEN Scanner

Într-adevăr, avem

$$D_{(a,\,b)}^{x_1x_2\ldots x_n}\cdot D_{(a)}^{x_1x_2\ldots x_n}=\left[D_{(a)}^{x_1x_2\ldots x_n}\cup D_{(b)}^{x_1x_2\ldots x_n}\right]\cdot D_{(a)}^{x_1x_2\ldots x_n}=$$

$$=D_{(a)}^{x_1x_2\ldots x_n}\cdot D_{(b)}^{x_1x_2\ldots x_n}\cup D_{(a)}^{x_1x_2\ldots x_n}\cdot D_{(a)}^{x_1x_2\ldots x_n}=$$

$$=D_{(a)}^{x_1x_2\ldots x_n}. \tag{3.18}$$

Din relația (3.18) mai rezultă și concluzia că un simbol de marcare nu se schimbă dacă-l înmulțim logic cu același simbol, conținînd însă numere de stare adăugate arbitrar.

Proprietățile conjuncției simbolurilor de marcare se dovedesc utile în transformarea schemelor cu contacte așa cum se va vedea în capitolul 9.

Exemplul 3.3. Să se verifice valabilitatea formulelor (3.16)—(3.18) pentru cazul unor funcții logice de două variabile binare.

Pentru a verifica valabilitatea formulelor (3.16) și (3.17), vom calcula valoarea următoarelor conjuncții logice:

$$D_{(1)}^{x_0x_1}\cdot D_{(3)}^{x_0x_1}=(\bar{x}_0x_1)\,(x_0x_1)=0,$$

$$D_{(2)}^{x_0x_1}\cdot D_{(2)}^{x_0x_1}=(x_0\bar{x}_1)\,(x_0\bar{x}_1)=x_0\bar{x}_1.$$

Valabilitatea unor relații de tipul (3.18) se poate verifica, de exemplu, considerînd conjuncția logică

$$D_{(0,\,1)}^{abc}\cdot D_{(1,\,3,\,5)}^{abc}=(\bar{a}\bar{b}\bar{c}\cup\bar{a}\bar{b}c)\,(\bar{a}\bar{b}c\cup\bar{a}bc\cup a\bar{b}c)=\bar{a}\bar{b}c=D_{(1)}^{abc}.$$

3.3. NEGAȚIA LOGICĂ

Ne propunem să găsim numerele de stare care corespund complementului unui simbol de marcare dat. În acest scop să considerăm conjuncția logică a unui simbol de marcare și a complementului lui:

$$D_N^{x_1 x_2 \dots x_n} \cdot \overline{D}_N^{x_1 x_2 \dots x_n} = 0. \tag{3.19}$$

Potrivit legii contradicției, valoarea acestei conjuncții trebuie să fie nulă. Să notăm

$$\overline{D}_N^{x_1 x_2 \dots x_n} = D_{N'}^{x_1 x_2 \dots x_n}, \tag{3.20}$$

în care caz relația (3.19) devine

$$D_N^{x_1 x_2 \dots x_n} \cdot D_{N'}^{x_1 x_2 \dots x_n} = 0. \tag{3.21}$$

Potrivit relației (3.16), egalitatea (3.21) este adevărată dacă numerele de ordine N și N' nu au nici un număr de stare comun. Relația (3.16) este sigur satisfăcută dacă N' cuprinde toate numerele de stare posibile necuprinse de N. Aceste numere de stare corespund biților 1 ai numărului N, exprimat în codul binar invers, adică lui $\overline{N}$, deci

$$N' = \overline{N}, \tag{3.22}$$

sau, aplicînd legea dublei negații, mai putem scrie

$$N = \overline{N}'. \tag{3.23}$$

Trecerea de la codul binar la codul binar invers se face negînd logic toți biții reprezentativi ai numărului exprimat în codul binar. În cazul nostru, trecerea la codul binar invers semnifică luarea în considerație a combinațiilor variabilelor de intrare care corespund valorii binare 0 a variabilei de ieșire.

Scanned with OKEN Scanner

Relaţia (3.21) este satisfăcută şi dacă N' cuprinde numai o parte dintre numerele de stare posibile necuprinse de N. În acest caz însă nu mai este satisfăcută relaţia (3.23), astfel încît nu putem lua în considerare această situaţie posibilă.

Folosind relaţia (3.22), relaţia (3.20) devine

$$\overline{D}_N^{x_1 x_2 \ldots x_n} = D_{\overline{N}}^{x_1 x_2 \ldots x_n}. \tag{3.24}$$

Aplicînd legea dublei negaţii, mai putem scrie

$$D_N^{x_1 x_2 \ldots x_n} = \overline{D}_{\overline{N}}^{x_1 x_2 \ldots x_n}. \tag{3.25}$$

Relaţiile (3.24) şi (3.25) exprimă proprietatea de complementaritate a simbolului de marcare. Remarcăm că aceste relaţii s-ar fi putut demonstra şi plecînd de la legea terţiului exclus.

Potrivit proprietăţii de complementaritate a simbolului de marcare, formula logică care corespunde unei scheme de comutaţie se poate deduce luînd în considerare fie valorile binare 1 ale variabilei de ieşire, fie valorile binare 0 ale acesteia şi negînd logic rezultatul obţinut. Această concluzie, cunoscută în literatura de specialitate [56, p. 51], îşi găseşte o justificare matematică prin consideraţiile de mai sus.

Facem observaţia

$$(\overline{N}) \neq (\overline{n}_1, \overline{n}_2, \ldots, \overline{n}_k), \tag{3.26}$$

întrucît printre numerele de stare negate se pot găsi unele care să nu corespundă valorii binare 0 a variabilei de ieşire, aşa cum cere definiţia lui $\overline{N}$.

Exemplul 3.4. Să se verifice valabilitatea proprietăţii de complementaritate a simbolului de marcare pentru cazul funcţiei logice date prin

$$F(X) = D_1^{x_0 x_1}.$$

Numărul de ordine $N = 1$ se exprimă în codul binar astfel:

$$N = 0.001$$

şi în codul binar invers prin

$$\overline{N} = \bar{0}.\bar{0}\bar{0}\bar{1} = 1.110.$$

Rezultă deci

$$\overline{N} = 16.$$

Formele operative ale numerelor de ordine N şi $\overline{N}$, care evidenţiază numerele de stare aferente lor, se găsesc folosind regulile expuse la exemplul 3.1. Acestea sînt

$$(N) = (3) \quad \text{şi} \quad (\overline{N}) = (0, 1, 2).$$

Astfel vom avea

$$\overline{D}_1^{x_0x_1} = \overline{D}_{(3)}^{x_0x_1} = \bar{x}_0\bar{x}_1 = \bar{x}_0 \cup \bar{x}_1,$$

$$D_{16}^{x_0x_1} = D_{(0,1,2)}^{x_0x_1} = \overline{x_0x_1} \cup \bar{x}_0x_1 \cup x_0\bar{x}_1 = \bar{x}_0 \cup \bar{x}_1,$$

$$D_1^{x_0x_1} = D_{(3)}^{x_0x_1} = x_0x_1,$$

$$\overline{D}_{16}^{x_0x_1} = \overline{D}_{(0,1,2)}^{x_0x_1} = \overline{\bar{x}_0 \cup \bar{x}_1} = x_0x_1.$$

Valabilitatea relaţiilor (3.24) şi (3.25) care exprimă proprietatea de complementaritate a simbolului de marcare, se verifică astfel prin egalităţile deduse din relaţiile de mai sus:

$$\overline{D}_1^{x_0x_1} = D_{16}^{x_0x_1} = D_{\bar{1}}^{x_0x_1},$$

$$D_1^{x_0x_1} = \overline{D}_{16}^{x_0x_1} = \overline{D}_{\bar{1}}^{x_0x_1}.$$

Facem observația

$$(\overline{N}) = (0, 1, 2) \neq (\overline{3}) = (0)$$

și

$$(\overline{\overline{N}}) = (3) \neq (\overline{0}, \overline{1}, \overline{2}) = (3, 2, 1).$$

Parantezele folosite în acest exemplu specifică forma operativă a numerelor de ordine.

Negarea logică a simbolului de marcare exprimă de fapt teoremele lui De Morgan.

Potrivit acestor teoreme, negarea logică a unei disjuncții de variabile binare este echivalentă cu conjuncția acestor variabile negate și negarea logică a unei conjuncții de variabile binare este echivalentă cu disjuncția acestor variabile negate.

În formă generală, ele se exprimă prin formulele

$$\overline{\bigcup_{k=1}^{n} x_k} = \prod_{k=1}^{n} \overline{x}_k \tag{3.27}$$

și

$$\overline{\prod_{k=1}^{n} x_k} = \bigcup_{k=1}^{n} \overline{x}_k. \tag{3.28}$$

Simbolul de marcare D (relația (1.3)) înlocuiește în fapt operatorul de disjuncție logică $\bigcup_{k=1}^{n}$, iar simbolul de marcare C (relația 3.4) înlocuiește operatorul generic de conjuncție logică $\prod_{k=1}^{n}$ pentru cazul cînd locul variabilelor x_k este luat de conjuncțiile, respectiv disjuncțiile variabilelor de intrare ale schemei de comutație corespunzînd combinaților pentru care variabila de ieșire are valoarea binară 1 sau 0.

Dacă mai avem în vedere faptul că negarea valorilor binare ale variabilelor de intrare care intră într-o combinație este echi-

valentă cu negarea numărului de stare aferent acestei combinații, adică cu numărul exprimat în codul binar invers, putem scrie

$$\overline{D}^{x_1 x_2 \ldots x_n}_{(n_1, n_2, \ldots, n_k)} = C^{x_1 x_2 \ldots x_n}_{(\overline{n}_1, \overline{n}_2, \ldots, \overline{n}_k)} \tag{3.29}$$

și

$$\overline{C}^{x_1 x_2 \ldots x_n}_{(n_1, n_2, \ldots, n_k)} = D^{x_1 x_2 \ldots x_n}_{(\overline{n}_1, \overline{n}_2, \ldots, \overline{n}_k)}. \tag{3.30}$$

Rezultă deci că formulele (3.29) și (3.30) exprimă teoremele lui De Morgan (3.27) și (3.28) pentru cazul cînd locul variabilelor x_k este luat de disjuncții sau conjuncții de mai multe variabile.

Remarcăm că simbolurile C și D din formulele (3.29) și (3.30) se referă la aceleași numere de stare. Aceasta implică luarea în considerare fie a valorii 1 a variabilei de ieșire, fie a valorii 0 a acesteia. Notațiile din formulele (3.29) și (3.30) se referă la valorile 1 ale variabilei de ieșire.

Folosind notațiile pentru valorile 0 ale variabilei de ieșire, vom putea scrie de asemenea

$$\tilde{D}^{x_1 x_2 \ldots x_n}_{(n_i, \ldots, n_s)} = C^{x_1 x_2 \ldots x_n}_{(\overline{n}_i, \ldots, \overline{n}_s)}. \tag{3.31}$$

și

$$\overline{C}^{x_1 x_2 \ldots x_n}_{(n_i, \ldots, n_s)} = D^{x_1 x_2 \ldots x_n}_{(\overline{n}_i, \ldots, \overline{n}_s)}. \tag{3.32}$$

Comparînd formula (3.31) cu formula (3.5), putem scrie

$$\overline{D}^{x_1 x_2 \ldots x_n}_{(n_i, n_h, \ldots, n_s)} = D^{x_1 x_2 \ldots x_n}_{(n_1, n_2, \ldots, n_j)}. \tag{3.33}$$

Avînd în vedere formula (3.24), mai rezultă o relație evidentă

$$(n_i, \ldots, n_s) = (n_1, \ldots, n_j), \tag{3.34}$$

potrivit căreia rezultă că prin luarea în considerare a valorii 0 a variabilei de ieșire se găsesc numerele de stare $(n_i, \ldots, n_j)$.

Exemplul 3.5. Să se verifice valabilitatea formulelor (3.29) și (3.30) pentru funcția logică dată de simbolul

$$F(X) = D^{x_1x_2x_3}_{(0,\,5,\,7)}.$$

Explicitînd simbolul dat și aplicînd teoremele lui De Morgan (3.27) și (3.28), obținem

$$F(X) = \bar{x}_1\bar{x}_2\bar{x}_3 \cup x_1\bar{x}_2x_3 \cup x_1x_2x_3,$$

$$\overline{F}(X) = \overline{(\bar{x}_1\bar{x}_2\bar{x}_3)}\ \overline{(x_1\bar{x}_2x_3)}\ \overline{(x_1x_2x_3)} =$$

$$= (x_1 \cup x_2 \cup x_3)(\bar{x}_1 \cup x_2 \cup \bar{x}_3)(\bar{x}_1 \cup \bar{x}_2 \cup \bar{x}_3) =$$

$$= C^{x_1x_2x_3}_{(7,\,2,\,0)} = C^{x_1x_2x_3}_{(\bar{0},\,\bar{5},\,\bar{7})}.$$

Rezultatul obținut corespunde formulei (3.29).

De asemenea avem

$$\overline{C}^{x_1x_2x_3}_{(\bar{0},\,\bar{5},\,\bar{7})} = \overline{(x_1 \cup x_2 \cup x_3)(\bar{x}_1 \cup x_2 \cup \bar{x}_3)(\bar{x}_1 \cup \bar{x}_2 \cup \bar{x}_3)} =$$

$$= \bar{x}_1\bar{x}_2\bar{x}_3 \cup x_1\bar{x}_2x_3 \cup x_1x_2x_3 = D^{x_1x_2x_3}_{(0,\,5,\,7)} = D^{\bar{\bar{x}}_1\bar{\bar{x}}_2\bar{\bar{x}}_3}_{(\bar{0},\,\bar{5},\,\bar{7})}.$$

Rezultatul obținut corespunde formulei (3.30).

Prin aplicarea legii dublei negații formulelor (3.29) și (3.30) se mai obține

$$D^{x_1x_2\ldots x_n}_{(n_1,\ldots n_k)} = \overline{C}^{x_1x_2\ldots x_n}_{(\bar{n}_1,\ldots \bar{n}_k)} \tag{3.35}$$

și

$$C^{x_1x_2\ldots x_n}_{(n_1,\ldots n_k)} = \overline{D}^{x_1x_2\ldots x_n}_{(\bar{n}_1,\ldots \bar{n}_k)}. \tag{3.36}$$

Scanned with OKEN Scanner

Bazaţi pe legea idempotenţei, putem exprima disjuncţia a două simboluri de marcare C printr-o formulă similară formulei (3.17):

$$C_{(a)}^{x_1 x_2 \ldots x_n} \cup C_{(b)}^{x_1 x_2 \ldots x_n} = C_{(a)}^{x_1 x_2 \ldots x_n} = C_{(b)}^{x_1 x_2 \ldots x_n}. \tag{3.37}$$

pentru cazul cînd $a = b$.

Dacă însă $a \neq b$, atunci, folosindu-ne de relaţia (3.36) şi de teoremele lui De Morgan, putem scrie

$$C_{(n_1, \ldots n_j)}^{x_1 x_2 \ldots x_n} \cup C_{(n_i, \ldots n_k)}^{x_1 x_2 \ldots x_n} = \overline{D}_{(\overline{n}_1, \ldots \overline{n}_j)}^{x_1 x_2 \ldots x_n} \cup \overline{D}_{(\overline{n}_i, \ldots \overline{n}_k)}^{x_1 x_2 \ldots x_n} =$$

$$= \overline{D_{(\overline{n}_1, \ldots \overline{n}_j)}^{x_1 x_2 \ldots x_n} \cdot D_{(\overline{n}_i, \ldots \overline{n}_j)}^{x_1 x_2 \ldots x_n}}. \tag{3.38}$$

O formulă similară se poate stabili pentru cazul disjuncţiei unor simboluri D şi C. Astfel putem scrie

$$C_{(n_1, \ldots n_j)}^{x_1 x_2 \ldots x_n} \cup D_{(n_i, \ldots n_k)}^{x_1 x_2 \ldots x_n} = \overline{D}_{(\overline{n}_1, \ldots \overline{n}_j)}^{x_1 x_2 \ldots x_n} \cup D_{(n_i, \ldots n_k)}^{x_1 x_2 \ldots x_n} =$$

$$= D_{(\overline{n}_1, \ldots \overline{n}_j)}^{x_1 x_2 \ldots x_n} \cup D_{(n_i, \ldots n_k)}^{x_1 x_2 \ldots x_n} = D_{[(\overline{n}_1, \ldots \overline{n}_j), (n_i, \ldots n_k)]}^{x_1 x_2 \ldots x_n}. \tag{3.39}$$

De asemenea, pentru conjuncţia unor astfel de simboluri obţinem

$$C_{(n_1, \ldots, n_j)}^{x_1 x_2 \ldots x_n} \cdot D_{(n_i, \ldots, n_k)}^{x_1 x_2 \ldots x_n} = \overline{D}_{(\overline{n}_1, \ldots, \overline{n}_j)}^{x_1 x_2 \ldots x_n} \cdot D_{(n_i, \ldots, n_k)}^{x_1 x_2 \ldots x_n} =$$

$$= D_{(\overline{n}_1, \ldots, \overline{n}_j)}^{x_1 x_2 \ldots x_n} \cdot D_{(n_i, \ldots, n_k)}^{x_1 x_2 \ldots x_n}. \tag{3.40}$$

Exemplul 3.6. Să se verifice valabilitatea relaţiilor (3.38)—(3.40) pentru cazul unor funcţii de trei variabile binare.

Pentru a verifica valabilitatea relaţiei (3.38) vom calcula expresia

$$C^{x_1x_2x_3}_{(0,\,5,\,7)} \cup C^{x_1x_2x_3}_{(1,\,5,\,2)} = \overline{\overline{(\bar{x}_1 \cup \bar{x}_2 \cup \bar{x}_3)\,(x_1 \cup \bar{x}_2 \cup x_3)\,(x_1 \cup x_2 \cup x_3)}}$$

$$\overline{\cup\,(\bar{x}_1 \cup \bar{x}_2 \cup x_3)\,(x_1 \cup \bar{x}_2 \cup x_3)\,(\bar{x}_1 \cup x_2 \cup \bar{x}_3)} =$$

$$= \overline{(x_1x_2x_3 \cup \bar{x}_1x_2\bar{x}_3 \cup \bar{x}_1\bar{x}_2\bar{x}_3)\,(x_1x_2\bar{x}_3 \cup \bar{x}_1x_2\bar{x}_3 \cup x_1\bar{x}_2x_3)} =$$

$$= \overline{\bar{x}_1x_2\bar{x}_3} = \bar{D}^{x_1x_2x_3}_{(2)}\,.$$

Mult mai expeditiv putem obţine acelaşi rezultat calculînd expresia cu ajutorul formulei (3.38):

$$\overline{D^{x_1x_2x_3}_{(0,\,5,\,7)} \cdot D^{x_1x_2x_3}_{(1,\,5,\,2)}} = \overline{D^{x_1x_2x_3}_{(7,\,2,\,0)} \cdot D^{x_1x_2x_3}_{(6,\,2,\,5)}} =$$

$$= \bar{D}^{x_1x_2x_3}_{(2)} = \bar{D}^{x_1x_2x_3}_{\overline{40}} = D^{x_1x_2x_3}_{\overline{40}} = D^{x_1x_2x_3}_{(0,\,1,\,3,\,4,\,5,\,6,\,7)}\,.$$

Pentru a verifica valabilitatea relaţiei (3.39) vom calcula expresia

$$C^{x_1x_2x_3}_{(0,\,5,\,7)} \cup D^{x_1x_2x_3}_{(1,\,5,\,2)} = (\bar{x}_1 \cup \bar{x}_2 \cup \bar{x}_3)\,(x_1 \cup \bar{x}_2 \cup x_3).$$

$$(x_1 \cup x_2 \cup x_3) \cup \bar{x}_1\bar{x}_2x_3 \cup x_1\bar{x}_2x_3 \cup \bar{x}_1x_2\bar{x}_3 = \bar{D}^{x_1x_2x_3}_{(0,\,7)}\,.$$

Mult mai expeditiv putem obţine acelaşi rezultat calculînd expresia cu ajutorul formulei (3.39)

$$D^{x_1x_2x_3}_{\overline{[(0,\,5,\,7),\,(1,\,5,\,2)]}} = D^{x_1x_2x_3}_{\overline{[(7,\,2,\,0),\,(1,\,5,\,2)]}} = D^{x_1x_2x_3}_{[(1,3,4,5,6),\,(1,5,2)]} =$$

$$= D^{x_1x_2x_3}_{(1,\,2,\,3,\,4,\,5,\,6)} = \bar{D}^{x_1x_2x_3}_{\overline{(1,\,2,\,3,\,4,\,5,\,6)}} = \bar{D}^{x_1x_2x_3}_{(0,7)}\,.$$

Scanned with OKEN Scanner

Pentru a verifica valabilitatea relaţiei (3.40) vom calcula expresia

$$C^{x_1x_2x_3}_{(0,5,7)} \cdot D^{x_1x_2x_3}_{(1,5,2)} = (\bar{x}_1 \cup \bar{x}_2 \cup \bar{x}_3)(x_1 \cup \bar{x}_2 \cup x_3).$$

$$(x_1 \cup x_2 \cup x_3)(\bar{x}_1\bar{x}_2x_3 \cup x_1\bar{x}_2x_3 \cup \bar{x}_1x_2\bar{x}_3) =$$

$$= \bar{x}_1\bar{x}_2x_3 \cup x_1\bar{x}_2x_3 = D^{x_1x_2x_3}_{(1,5)}.$$

Mult mai expeditiv putem obţine acelaşi rezultat calculînd expresia cu ajutorul formulei (3.40):

$$D^{x_1x_2x_3}_{(\bar{0},\bar{5},\bar{7})} \cdot D^{x_1x_2x_3}_{(1,5,2)} = D^{x_1x_2x_3}_{(7,2,0)} \cdot D^{x_1x_2x_3}_{(1,5,2)} =$$

$$= D^{x_1x_2x_3}_{(1,3,4,5,6)} \cdot D^{x_1x_2x_3}_{(1,5,2)} = D^{x_1x_2x_3}_{(1,5)}.$$

3.4. PERMUTABILITATEA INDICILOR

Indicele superior al simbolului de marcare poate fi supus la $n!$ permutări. În corespondenţă cu acestea se va modifica şi indicele inferior al simbolului.

Prin modificarea corespunzătoare a celor doi indici, expresia logică a schemei de comutaţie nu se va modifica, întrucît o expresie logică nu poate depinde de modul de ordonare a variabilelor binare de care depinde.

Rezultă deci că pe calea permutărilor corespunzătoare efectuate în cei doi indici ai simbolului de marcare se pot găsi scheme de comutaţie echivalente cu funcţii care să corespundă la simboluri de marcare diferite între ele numai ca urmare a modificării ordinii de scriere a variabilelor de intrare din indicele superior al acestora.

Scanned with OKEN Scanner

Exemplul 3.7. Să se găsească cele $n!$ $(n = 3)$ simboluri de marcare care corespund funcţiei logice date prin formula

$$F(A) = \bar{a}bc \cup a\bar{b}\bar{c}.$$

Primul simbol de marcare, care corespunde ordinii normale de scriere a variabilelor de intrare, este

$$F(A) = D^{abc}_{(3,4)} = D^{abc}_{30}.$$

Modificînd biţii reprezentativi ai numerelor de stare

$$(3)_2 = 011 \quad \text{şi} \quad (4)_2 = 100$$

în corespondenţă cu şirurile permutate ale variabilelor de intrare a, b şi c, se obţin restul celorlalte cinci simboluri de marcare care corespund aceleiaşi formule logice:

$$F(A) = D^{cab}_{(5,2)} = D^{cab}_{44} = c\bar{a}b \cup \bar{c}a\bar{b},$$

$$F(A) = D^{bca}_{(6,1)} = D^{bca}_{102} = bc\bar{a} \cup \bar{b}\bar{c}a,$$

$$F(A) = D^{acb}_{(3,4)} = D^{acb}_{30} = \bar{a}cb \cup a\bar{c}\bar{b},$$

$$F(A) = D^{cba}_{(6,1)} = D^{cba}_{102} = cb\bar{a} \cup \bar{c}\bar{b}a,$$

$$F(A) = D^{bac}_{(5,2)} = D^{bac}_{44} = b\bar{a}c \cup \bar{b}a\bar{c}.$$

Dacă însă permutăm numai variabilele din indicele superior al simbolului de marcare, lăsînd nepermutaţi biţii reprezentativi ai numerelor de stare din indicele inferior al acestuia sau procedăm invers, se vor obţine expresii logice diferite care corespund la scheme de comutaţie diferite.

Concluzia de mai sus apare evidentă dacă avem în vedere că, de fapt, permutarea efectuată cu variabilele din indicele su-

Scanned with OKEN Scanner

perior al simbolului de marcare sau cu biţii reprezentativi ai numerelor de stare din indicele inferior al acestuia este echivalentă cu o schimbare corespunzătoare de variabile în expresia logică exprimată prin acest simbol.

Expresia logică permutată se va obţine prin schimbarea variabilelor pe care le conţine după legea permutării efectuate fie în indicele superior, fie în cel inferior al simbolului de marcare.

Se pot stabili următoarele reguli de permutare :

1) în cazul permutării numai a variabilelor din indicele superior al simbolului de marcare, expresia logică corespunzătoare noului simbol se obţine prin permutarea după aceeaşi lege a variabilelor conţinute de expresia logică exprimată de simbolul iniţial;

2) în cazul permutării numai a biţilor reprezentativi ai numerelor de stare din indicele inferior al simbolului de marcare, expresia logică corespunzătoare noului simbol se obţine permutînd variabilele expresiei logice exprimate de simbolul iniţial după legea permutării inverse celei a biţilor numerelor de stare.

Ultima regulă se explică astfel : se presupune că iniţial am permutat ambii indici ai simbolului, expresia logică exprimată de acesta rămînînd neschimbată. Revenirea la vechiul indice superior se va face, evident, printr-o pemutare inversă.

Permutarea nesimultană a indicilor din simbolul de marcare oferă deci posibilitatea de a găsi simboluri de marcare noi pentru funcţii logice ce se pot obţine prin permutări.

Exemplul 3.8. Să se găsească simbolurile de marcare şi expresiile logice posibil de obţinut prin permutarea indicilor simbolului

$$F(Z) = D_{42}^{xyz} = y\bar{z}.$$

Se determină mai întîi forma operativă a indicelui inferior din simbolul de marcare. Pentru aceasta convertim în codul binar numărul de ordine N, care prin definiţie este dat în codul octal

$$(104)_2 = 01.000.100.$$

Biţii 1 corespund deci numerelor de stare (5) şi (1), astfel încît putem scrie

$$F(Z) = D_{(1,5)}^{xyz}.$$

Întrucît

$$n! = 3! = 6,$$

trebuie să existe încă cinci funcţii logice şi, corespunzătoare lor, cinci simboluri de marcare, care se pot obţine prin permutarea fie a indicelui superior, fie a biţilor reprezentativi ai numerelor de stare.

Prin permutări efectuate numai în indicele superior al simbolului de marcare şi folosind regula 1) expusă mai sus, obţinem

$$F_1(Z) = D_{104}^{zxy} = x\bar{y},$$

$$F_2(Z) = D_{104}^{yzx} = z\bar{x},$$

$$F_3(Z) = D_{104}^{xzy} = z\bar{y},$$

$$F_4(Z) = D_{104}^{zyx} = y\bar{x},$$

$$F_5(Z) = D_{104}^{yxz} = x\bar{z}.$$

Aceleaşi funcţii logice se obţin prin permutări efectuate numai în indicele inferior al simbolului, folosind regula 2) dată mai sus :

$$D_{(1,\,3)}^{xyz} = D_{120}^{xyz} = z\bar{x} = F_2(Z),$$

$$D_{(4,\,5)}^{xyz} = D_{14}^{xyz} = x\bar{y} = F_1(Z),$$

$$D_{(1,\,5)}^{xyz} = D_{104}^{xyz} = z\bar{y} = F_3(Z),$$

$$D_{(2,\,3)}^{xyz} = D_{60}^{xyz} = y\bar{x} = F_4(Z),$$

$$D_{(4,\,6)}^{xyz} = D_{12}^{xyz} = x\bar{z} = F_5(Z).$$

Scanned with OKEN Scanner

3.5. SCOATEREA ÎN FACTOR

Analiza tabelei de adevăr sau a numerelor binare corespunzătoare combinaţiilor variabilelor de intrare care definesc o schemă de comutaţie ne poate sugera ideea scoaterii în factor a unor variabile ce intră în indicele superior al simbolului de marcare. Aceasta devine posibil atunci cînd un grup sau toate numerele de stare din simbol au comună valoarea binară 1 sau 0 pentru una sau pentru mai multe variabile de intrare.

Micşorarea numărului de variabile din indicele superior al simbolului de marcare uşurează mult sinteza expresiei logice minimale aferente tabelei de adevăr sau simbolului de marcare respectiv.

Fie, de exemplu, o schemă de comutaţie definită prin simbolul de marcare scris în forma sa operativă dezvoltată dată de formula (3.7).

Presupunem că tabela de adevăr după care s-a stabilit acest simbol conţine în coloanele $x_1, x_2, \ldots, x_j$ numai valori 0, care sînt astfel comune tuturor numerelor de stare. Se presupune că liniile tabelei de adevăr sînt numerotate în ordinea crescătoare a numerelor de stare sau, dacă funcţia logică este definită printr-un număr mai mic de numere de stare decît cel maxim posibil, acestea sînt considerate pe linii în ordinea lor crescătoare.

În ipoteza făcută, disjuncţia logică considerată admite ca factor comun conjuncţia variabilelor comune cu valoarea negată, astfel încît putem scrie

$$D^{x_1 x_2 \ldots x_n}_{(n_1, \ldots, n_k)} = \bar{x}_1 \bar{x}_2 \ldots \bar{x}_j D^{x_{j+1} x_{j+2} \ldots x_n}_{(n_1, \ldots, n_k)} . \tag{3.41}$$

În acest caz, numerele de stare nu s-au modificat prin scoaterea lor ca factor, întrucît biţii 1 ai acestora se refereau numai la variabilele ne scoase ca factor.

Dacă valoarea binară 1 este comună variabilelor $x_{j+1}, x_{j+2}, \ldots, x_n$, un raţionament analog celui de mai sus ne conduce la formula

$$D^{x_1 x_2 \ldots x_n}_{(n_1, \ldots, n_k)} = x_{j+1} x_{j+2} \ldots x_n D^{x_1 x_2 \ldots x_j}_{(n'_1, \ldots, n'_k)} . \tag{3.42}$$

În acest caz, variabilele scoase în factor au valoarea nenegată, iar numerele de stare s-au modificat prin scoaterea lor în factor întrucît biţii 1 ai acestora se refereau şi la variabilele scoase în factor.

Scanned with OKEN Scanner

Formulele (3.41) şi (3.42) se pot aplica şi simultan aceluiaşi simbol de marcare, prin aceasta putîndu-se scoate factor valorile comune 1 şi 0 ale unui grup sau a tuturor numerelor de stare.

Exemplul 3.9. Să se descompună simbolul de marcare

$$F(X) = D^{x_0x_1x_2x_3}_{(4,5,7,10,11,13,16,17)}$$

în doi termeni care să permită scoaterea în factor a unui număr maxim de variabile.

Pentru uşurarea selectării variabilelor ce vor intra în cei doi termeni este utilă reprezentarea unei coloane care să conţină numerele de stare ce definesc simbolul de marcare în ordinea lor crescătoare şi în codul binar (extras din tabelă de adevăr).

În cazul nostru, folosind numai indicele inferior al simbolului de marcare dat, putem forma următoarea coloană:

0.100
0.101
0.111
1.000
1.001
1.011
1.110
1.111

Examinarea acestei coloane evidenţiază posibilitatea optimă de a forma un termen din primele trei numere de stare (linii ale coloanei), care admit ca factor comun conjuncţia $\bar{x}_0x_1$, şi alt termen, conţinînd restul numerelor de stare, care admite ca factor comun numai variabila x_0. Se obţine astfel

$$F(X) = \bar{x}_0x_1D^{x_2x_3}_{(0,1,3)} \cup x_0D^{x_1x_2x_3}_{(0,1,3,6,7)}.$$

Noile numere de stare rezultă direct din coloana pe care am folosit-o pentru scoaterea în factor.

În formă neoperativă, transformarea efectuată se scrie astfel:

$$D^{x_0x_1x_2x_3}_{6.723} = \bar{x}_0x_1D^{x_2x_3}_{15} \cup x_0D^{x_1x_2x_3}_{323}.$$

Scanned with OKEN Scanner

Capitolul 4 GRAFUL ASOCIAT UNUI SIMBOL DE MARCARE

Informaţiile cuprinse de un simbol de marcare pot fi redate sub forma unui graf. Graful asociat unui simbol de marcare evidenţiază grupările de numere de stare adiacente cuprinse de simbol, grupări care permit în final explicitarea simbolului de marcare printr-o expresie logică minimală.

Referindu-ne la reprezentarea numerelor de stare în codul binar, două astfel de numere se numesc adiacente dacă ele nu diferă decît prin biţii reprezentativi ai unui singur rang. Pentru un număr reprezentat în codul binar prin n biţi, sînt deci posibile n adiacenţe.

Întrucît numerele de stare sînt scrise în codul octal, se poate uşor deduce că două astfel de numere sînt adiacente dacă diferenţa dintre numărul reprezentat în codul binar prin k biţi 1 şi cel reprezentat prin $k-1$ biţi 1 este 1,2,4,10,20,40,100, 200, 400, 1000,...etc. Aceste numere, numite „diferenţe de adiacenţă", sînt egale cu valoarea în codul octal a bitului reprezentativ al unui număr dat în codul binar, număr ce conţine un singur bit 1.

Exemplul 4.1. Să se scrie o succesiune de şase numere adiacente reprezentate în codul binar şi în cel octal.

Vom lua ca exemplu numere cu şase biţi reprezentativi în codul binar. Potrivit definiţiei adiacenţei, fiecare număr diferă de numerele vecine din şir prin valoarea binară a unui singur bit. Astfel, în codul binar putem scrie următorul şir de numere adiacente :

000.000,010.000,011.000,011.001,011.011,011.111.

În codul octal, aceste numere se scriu astfel :

0,20,30,31,33,37.

Scanned with OKEN Scanner

Observăm că diferențele de adiacență corespund respectiv la

$$20,10,1,2,4.$$

Putem acum defini graful aferent unui simbol de marcare. Pentru un simbol de marcare ce conține n variabile de intrare se trasează $n+1$ nivele orizontale (fig. 4.1).

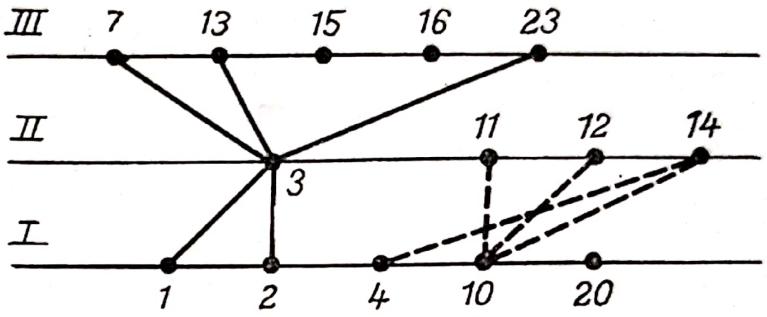


Fig. 4.1. — Exemplu de reprezentare a nodurilor și ramurilor unui graf.

Ele corespund la numere de stare care conțin $0,1,2,\ldots,n$, biți 1.

Pe fiecare nivel se vor reprezenta numerele de stare ce au același număr de biți 1. Aceste puncte reprezintă nodurile grafului. Lîngă fiecare nod al grafului se notează numărul de stare aferent lui.

Pentru identificarea mai ușoară a numerelor de stare ce trebuie reprezentate pe aceste nivele mai ales atunci cînd încă nu este formată o rutină în acest sens, se poate atașa o coloană tabelei de adevăr, ca în tabela 1.1., în care se trece numărul de biți 1 ai fiecărui număr de stare.

Prin modul de reprezentare adoptat, nodurile adiacente, corespunzătoare numerelor de stare adiacente, se vor găsi printre nodurile situate pe nivele vecine.

Spre exemplu, nodul (3) din figura 4.1., reprezentat pe nivelul II, corespunzător celor doi biți 1 pe care-i conține numărul de stare $(3)_2=11$, va fi vecin cu nodurile adiacente (7), (13) și (23), situate pe nivelul III, care corespunde celor trei biți „1" conținuți de numerele de stare $(7)_2=111$, $(13)_2=1.011$ și $(23)_2=10.011$ și cu nodurile adiacente (2) și (1), situate pe nivelul I, care corespunde unicului bit 1 reprezentativ al numerelor de stare $(2)_2=10$ și $(1)_2=1$.

Pe nivelul I se mai găsesc însă şi nodurile (4), (10), (20),... care nu sînt adiacente cu nodul (3), întrucît ele nu satisfac definiţia de adiacenţă, şi anume

$$(3) - (4) = -1, (3) - (10) = -7, (3) - (20) = -13, \ldots \text{ etc.}$$

Aceste noduri vor fi însă adiacente cu alte noduri situate pe nivelul II, de exemplu (4) cu (14), (10) cu (11), cu (12) cu (14) etc.

Dreapta care uneşte două noduri adiacente reprezintă o ramură a grafului.

Fiecare ramură a grafului se caracterizează prin variabila care dispare prin disjuncţia conjuncţiilor reprezentate de nodurile unite. Această variabilă corespunde cu poziţia bitului 1 reprezentativ al numărului diferenţă de adiacenţă (exprimat în codul octal) dintre numerele de stare aferente nodurilor unite.

Spre exemplu, ramura grafului care uneşte nodurile (17) şi (15) este caracterizată prin variabila x_2 din şirul $x_0x_1x_2x_3$, care reprezintă indicele superior al simbolului de marcare, variabilă ce dispare prin disjuncţia conjuncţiilor reprezentate de nodurile unite.

Într-adevăr, în acest caz avem

$$x_0x_1x_2x_3 \cup x_0x_1\bar{x}_2x_3 = x_0x_1x_3(x_2 \cup \bar{x}_2) = x_0x_1x_3.$$

În cazul examinat, locul variabilei dispărute prin disjuncţie în şirul variabilelor de intrare din indicele superior al simbolului de marcare corespunde cu bitul 1 reprezentativ al numărului 2 exprimat în codul octal, numărul 2 fiind egal cu numărul diferenţă de adiacenţă dintre numerele de stare (17) şi (15).

Prin disjuncţia a două combinaţii adiacente de variabile se modifică şi simbolul de marcare. Astfel, din indicele superior al simbolului de marcare dispare variabila care caracterizează ramura respectivă a grafului, iar în indicele inferior, scris în formă operativă, locul celor două numere de stare adiacente este luat de un singur număr de stare. Acesta se poate găsi uşor scriind în codul binar unul dintre cele două numere de stare unite de ramura grafului şi tăind (prin subliniere) bitul reprezentativ care corespunde variabilei dispărute din indicele superior al simbolului de marcare, adică variabilei dispărute prin disjuncţie.

Scanned with OKEN Scanner

Exemplul 4.2. Să se construiască graful aferent simbolului de marcare

$$F(C) = D^{x_0x_1x_2x_3}_{(0,10,13,17)}$$

și să se găsească forma modificată a simbolului ca urmare a adiacențelor posibile între numerele de stare.

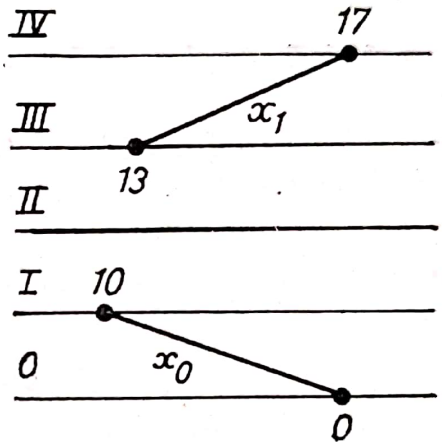


Fig. 4.2. — Graful aferent exemplului 4.2.

În graful redat în figura 4.2, prin cifre romane au fost notate nivelele care corespund cu numărul de biți 1 ale nodurilor grafului.

Graful reprezentat în această figură evidențiază două adiacențe, și anume

$$(17) - (13) = 4 \quad \text{și} \quad (10) - (0) = 10.$$

Rezultă deci că variabilele care caracterizează cele două ramuri ale grafului vor corespunde respectiv bitului reprezentativ al numărului octal 4 și al numărului octal 10, din șirul $x_0x_1x_2x_3$, adică respectiv variabilelor x_1 și x_0 din acest șir. Acestea sînt variabilele care dispar prin disjuncții.

Pentru a găsi numerele de stare care înlocuiesc în simbolul de marcare numerele de stare adiacente, scriem în codul binar cîte un singur reprezentant din aceste grupe, de exemplu

$$(17)_2 = 1.\underline{1}11 = 111 \quad \text{și} : \quad (0)_2 = \underline{0}.000 = 000,$$

tăind biţii corespunzători (cei subliniaţi mai sus) numerelor octale 4 şi 10. Se găsesc astfel noile numere de stare :

$$(111)_8 = 7 \quad \text{şi} \quad (000)_8 = 0.$$

Ele corespund într-adevăr combinaţiilor $x_0x_2x_3$ şi $\bar{x}_1\bar{x}_2\bar{x}_3$, rămase în urma disjuncţiilor prin care se explicitează simbolul de marcare dat :

$$x_0x_1x_2x_3 \cup x_0\bar{x}_1x_2x_3 = x_0x_2x_3(x_1 \cup \bar{x}_1) = x_0x_2x_3,$$

$$x_0\bar{x}_1\bar{x}_2\bar{x}_3 \cup \bar{x}_0\bar{x}_1\bar{x}_2\bar{x}_3 = \bar{x}_1\bar{x}_2\bar{x}_3\,(x_0 \cup \bar{x}_0) = \bar{x}_1\bar{x}_2\bar{x}_3.$$

Prin folosirea formulei (3.7) se ajunge final la următoarele două simboluri de marcare prin care se explicitează simbolul dat :

$$F(C) = D^{x_0x_1x_2x_3}_{(13,17)} \cup D^{x_0x_1x_2x_3}_{(0,10)} = D^{x_0x_2x_3}_{(7)} \cup D^{x_1x_2x_3}_{(0)}.$$

În cazul mai multor adiacenţe, ramurile grafului formează în plan contururi poligonale, care, privite spaţial, apar ca poliedre. Printr-o amplasare corespunzătoare a nodurilor adiacente, graful apare sub forma unor figuri geometrice regulate.

Cel mai simplu contur poligonal dintr-un graf este un patrulater. Patrulaterul se va numi simetric dacă are laturile opuse caracterizate prin aceeaşi variabilă dispărută prin disjuncţia nodurilor unite.

Cel mai simplu poliedru dintr-un graf este un hexaedru. Hexaedrul se va numi simetric dacă are drept feţe patrulatere simetrice. Se pot astfel defini poliedre simetrice cu mai mult de şase feţe.

Patrulaterele sau poliedrele simetrice reunesc grupări de cîte 2^k ($k = 1,2,3,\ldots$) noduri adiacente.

Vom numi „graf complet", graful care conţine toate cele 2^n noduri corespunzătoare tuturor numerelor de stare posibile pentru cele n variabile de intrare.

În figură 4.3 s-au dat pentru exemplificare grafurile complete pentru una, două, trei patru şi cinci variabile de intrare.

O funcţie logică definită printr-un simbol de marcare care conţine toate numerele de stare posibile pentru cele n variabile

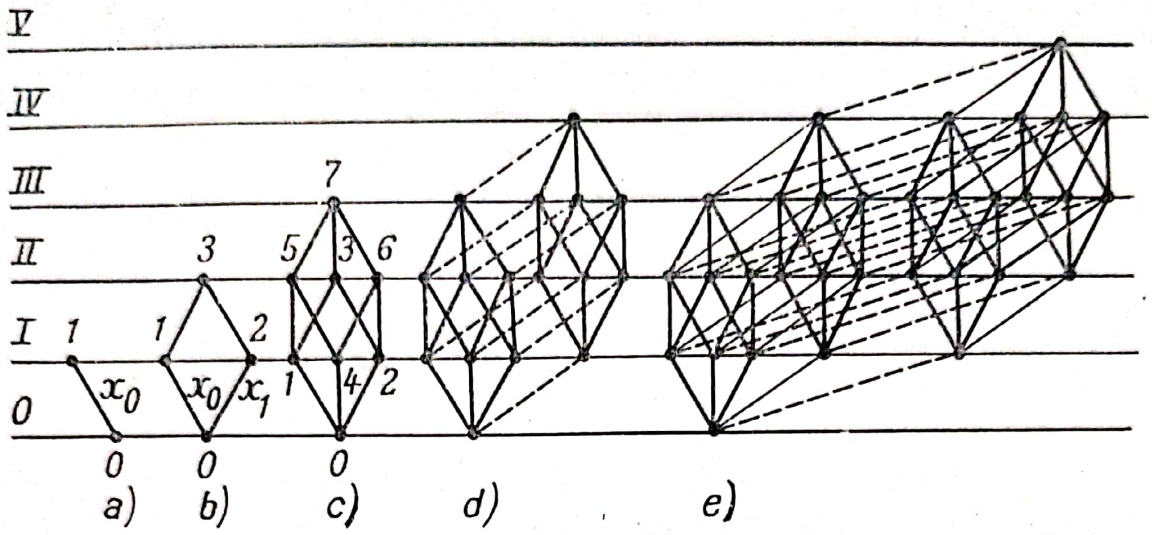


Fig. 4.3. — Grafuri complete de :
a) una *b*) două *c*) trei *d*) patru *e*) cinci variabile binare.

de intrare este reprezentată printr-un graf complet. Ea corespunde funcției logice „constanta 1", adică simbolului de marcare

$$D^{x_1 x_2 \ldots x_n}_{(2^{2^n}-1)_8} = 1. \tag{4.1}$$

Dacă graful complet se construiește pentru valorile 0 ale variabilei de ieșire, atunci trebuie luate în considerare numerele de stare care corespund acestei valori a variabilei de ieșire, obținîndu-se valoarea negată a funcției logice. În acest caz, avînd în vedere relația (3.24), putem scrie

$$\bar{D}^{x_1 x_2 \ldots x_n}_{(2^{2^n}-1)_8} = D^{x_1 x_2 \ldots x_n}_{(2^{2^n}-1)_8} = D^{x_1 x_2 \ldots x_n}_{0} = 0. \tag{4.2}$$

Rezultatul obținut corespunde funcției logice „constanta 0".

Exemplul 4.3. Să se verifice valabilitatea relațiilor (4.1) și (4.2) pentru cazul unei funcții logice de două variabile binare.

Relaţia (4.1) este verificată prin

$$D^{x_1x_2}_{(2^{2^2}-1)_8} D^{x_1x_2}_{17} = D^{x_1x_2}_{(0,1,2,3)} = \bar{x}_1\bar{x}_2 \cup \bar{x}_1x_2 \cup x_1\bar{x}_2 \cup x_1x_2 =$$

$$= \bar{x}_1(\bar{x}_2 \cup x_2) \cup x_1(\bar{x}_2 \cup x_2) = \bar{x}_1 \cup x_1 = 1,$$

iar relaţia (4.2) prin

$$D^{x_1x_2}_{\overline{(2^{2^2}-1)_8}} = D^{x_1x_2}_0 = 0.$$

Dacă dintre cele 2^{2^n} funcţii logice posibile pentru n variabile de intrare eliminăm funcţiile logice „constanta 0" şi „constanta 1", mai rămîn $(2^{2^n}-2)$ funcţii logice. Acestea vor fi reprezentate de „grafuri incomplete", adică de grafuri care nu conţin decît o parte a celor 2^{2^n} noduri posibile.

Un graf incomplet poate fi format din noduri singulare, grupări de 2,4,8,.... sau în general de 2^{2^k} $(k<n)$ noduri, ori de asociaţii de noduri singulare şi grupări de noduri adiacente între ele.

Fiecare nod singular sau grupare de noduri adiacente dintr-un graf incomplet reprezintă un termen al expresiei logice prin care se explicitează un simbol de marcare.

Adiacenţele posibile între nodurile grupate fac ca din indicele superior al simbolului de marcare, care corespunde unui termen al expresiei logice exprimate de acesta, să dispară variabilele care caracterizează ramurile grafului.

Modificarea indicelui inferior al acestui simbol se găseşte scriind în codul binar un reprezentant dintre nodurile grupate şi tăind (prin subliniere) biţii corespunzători poziţiilor variabilelor dispărute din indicele superior. Numărul de stare obţinut astfel înlocuieşte tot grupul numerelor de stare corespunzătoare nodurilor unite prin grupare.

Exemplul 4.4. Să se expliciteze simbolul de marcare

$$F(A) = D^{x_0x_1x_2x_3}_{(0,3,7,11,13,15,17)},$$

folosind graful incomplet aferent lui.

Prin gruparea nodurilor adiacente rezultă graful dat în figura 4.4. Configurația lui evidențiază existența a trei termeni, care în formă simbolică sînt descriși de formula

$$F(A) = D^{x_0x_1x_2x_3}_{(0)} \cup D^{x_0x_1x_2x_3}_{(11,13,15,17)} \cup D^{x_0x_1x_2x_3}_{(3,7,13,17)}.$$

Diferențele dintre numerele de stare adiacente

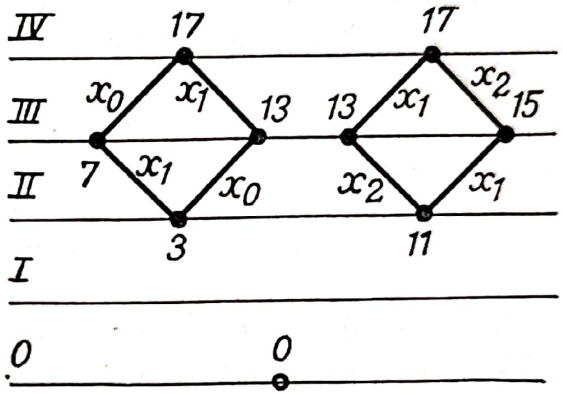


Fig. 4.4. — Graful incomplet aferent exemplului 4.4.

$$(15) - (11) = 4 \qquad (17) - (13) = 4$$

$$(7) - (3) = 4 \quad \text{și} \quad (17) - (7) = 10$$

$$(13) - (3) = 10 \qquad (13) - (11) = 2$$

$$(17) - (15) = 2$$

ne dau variabilele care caracterizează ramurile grafului.

Primul patrulater simetric din acest graf are laturile opuse (3,7) și (13,17), caracterizate prin variabila x_1, și laturile opuse (3,13) și (7,17), caracterizate prin variabila x_0. Aceste variabile ocupă respectiv locurile biților 1 reprezentativi ai numerelor octale diferență de adiacență 4 și 10 în șirul variabilelor de intrare $x_0x_1x_2x_3$.

Al doilea patrulater simetric din acest graf are laturile opuse (11,13) și (15,17), caracterizate prin variabila x_2, și laturile opuse (11,15) și (13,17), caracterizate prin variabila x_1. Aceste variabile ocupă respectiv locurile biților 1 reprezentativi ai numerelor octale 2 și 4 în șirul variabilelor de intrare $x_0x_1x_2x_3$.

Considerăm drept reprezentanți ai celor două grupări ale grafului nodurile (7) și (11), ale căror numere de stare se scriu în codul binar astfel:

$$(7)_8 = (\underline{0}.\underline{1}11)_2 = (11)_2 = (3)_8 \quad \text{și:} \quad (11)_8 = (1.\underline{00}1)_2 = (11)_2 = (3)_8.$$

Tăind biții corespunzători, respectiv, variabilelor x_1 și x_2, dispărute prin disjuncții pentru nodul (11) și cei corespunzători variabilelor x_0 și x_1 pentru nodul (7), anume biții subliniați mai sus, se obțin noile numere de stare ale termenilor (3) pentru ambii termeni. Astfel putem scrie

$$F(A) = D_{(0)}^{x_0x_1x_2x_3} \cup D_{(3)}^{x_0x_3} \cup D_{(3)}^{x_2x_3} =$$

$$= \bar{x}_0\bar{x}_1\bar{x}_2\bar{x}_3 \cup x_0x_3 \cup x_2x_3 = \bar{x}_0\bar{x}_1\bar{x}_2\bar{x}_3 \cup x_3(x_0 \cup x_2).$$

Prin modul de definire și de construcție a grafului asociat simbolului de marcare, acesta se pretează a fi folosit pentru minimizarea expresiilor booleene și, mai ales, ne oferă regulile după care se pot găsi grupele de adiacență necesare operațiilor de minimizare. Aceste reguli se vor expune în capitolul 6 al acestei lucrări.

Scanned with OKEN Scanner

Capitolul 5 CLASIFICAREA FUNCȚIILOR LOGICE

Simbolul de marcare (1.2) încadrează toate funcțiile logice posibile pentru un număr dat al variabilelor de intrare. Numărul acestora pentru n variabile binare de intrare este dat de 2^{2^n} [82, p. 15].

După cum s-a arătat la paragraful 3.4, unei funcții logice date îi pot corespunde mai multe simboluri de marcare, care se pot găsi permutînd după aceeași lege indicele superior al simbolului și biții reprezentativi ai numerelor de stare.

Pentru a determina univoc o funcție logică, printr-un simbol de marcare unic, se impune deci un criteriu de selecție. În acest scop vom lua în considerare numărul de ordine de valoare minimă. Pentru a-l obține vom folosi regulile de permutare stabilite în paragraful 3.4. Astfel se găsește mai întîi numărul biților 1 conținuți de toate numerele de stare care definesc simbolul, corespunzătoare pozițiilor fiecărei variabile din indicile superior. Ordonînd apoi aceste numere în ordinea crescătoare de la stînga spre dreapta se găsesc permutările ce trebuie efectuate în scopul obținerii unui număr de ordine minim. Se efectuează apoi aceste permutări după regulile amintite, găsindu-se astfel simbolul de marcare care corespunde unui număr de ordine minim.

Pentru determinarea univocă a unei funcții logice printr-un simbol de marcare se mai impune discriminarea funcțiilor logice reductibile la un număr de variabile mai mic decît cel conținut de indicele superior al simbolului.

Astfel, dacă toate numerele de stare care definesc simbolul de marcare formează o grupare de numere de stare adiacente între ele, grupare ce se poate reprezenta printr-un graf complet, funcția logică respectivă depinde de mai puține variabile de intrare decît cele conținute de indicele superior al simbolului. Noul simbol de marcare se deduce în acest caz după regulile stabilite în capitolul 4 sau, mai simplu, după cele ce se vor stabili în capitolul 6.

Scanned with OKEN Scanner

Consideraţiile de mai sus ne permit să enunţăm următorul criteriu de clasificare a funcţilor logice: o funcţie logică de mai multe variabile binare este definită univoc printr-un simbol de marcare ireductibil ca număr de variabile şi caracterizat prin numărul de ordine minim posibil. Vom denumi acest simbol „simbol de marcare tipic".

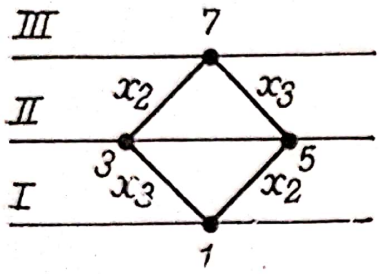


Fig. 5.1. — Graful complet aferent exemplului 5.1.

Acest criteriu de clasificare permite identificarea schemelor de comutaţie după simbolul de marcare tipic aferent lor. Astfel, referindu-ne la o schemă de comutaţie, este suficient să dăm numai simbolul de marcare tipic care o caracterizează complet, exceptînd doar temporizările programate ce se manifestă în funcţionarea ei.

Exemplul 5.1. Să se determine simbolurile de marcare care caracterizează univoc funcţiile logice descrise de

$$F(X) = D^{x_1x_2x_3x_4}_{(1,\,3,\,5,\,7)} \quad \text{şi:} \quad F(Y) = D^{x_1x_2x_3}_{(0,\,4,\,6)}.$$

Construind graful asociat simbolului de marcare al funcţiei logice $F(X)$, aşa cum se vede în figura 5.1, se constată că toate numerele de stare care caracterizează acest simbol sînt cuprinse de un graf complet. Rezultă deci că funcţia logică $F(X)$ este reductibilă.

Folosind regulile date în capitolul 4, găsim noul simbol de marcare

$$F(X) = D^{x_1x_4}_{(1)} = D^{x_1x_4}_{4}.$$

Numărul de ordine $N = 4$ obţinut astfel nu este cel minim posibil, deci simbolul de marcare obţinut nu este tipic.

Astfel, permutînd variabilele x_1 şi x_4 între ele, noul număr de stare obţinut prin permutarea

$$01 \rightarrow 10$$

este (2), rezultînd în acest caz simbolul tipic de marcare

$$D^{x_4x_1}_{(2)} = D^{x_4x_1}_{2}.$$

Cele două simboluri deduse descriu într-adevăr aceeași funcție logică, întrucît

$$D_4^{x_1x_4} = \bar{x}_1x_4 \quad \text{și} \quad D_2^{x_4x_1} = x_4\bar{x}_1.$$

Pentru construcția unui graf complet este nevoie de un număr cu soț de numere de stare, și anume $2,4,8,\ldots,2^n$. Întrucît simbolul de marcare pentru funcția logică $F(Y)$ conține un număr de numere de stare fără soț, această funcție logică este ireductibilă, în sensul că nu dispar variabile.

Prin procedeul indicat în exemplul 1.1, se găsește că

$$F(Y) = D_{2120}^{x_1x_2x_3}.$$

Pentru a obține un număr de ordine N de valoare octală minimă se consideră următoarea ordine de scriere a variabilelor de intrare :

$$x_3x_2x_1,$$

această ordine se găsește prin încercări.

Numerele de stare care corespund simbolului de marcare tipic se obțin deci prin permutările

$$000 \to 000, \qquad 100 \to 001 \qquad \text{și} \qquad 110 \to 011.$$

Simbolul tipic de marcare al funcției logice $F(Y)$ va fi deci

$$F(Y) = D_{(0,\,1,\,3)}^{x_3x_2x_1} = D_{320}^{x_3x_2x_1}.$$

Cele două simboluri deduse pentru $F(Y)$ sînt echivalente, întrucît

$$D_{(0,\,4,\,6)}^{x_1x_2x_3} = \bar{x}_1\bar{x}_2\bar{x}_3 \cup x_1x_2\bar{x}_3 \cup x_1x_2\bar{x}_3$$

și

$$D_{(0,\,1,\,3)}^{x_3x_2x_1} = \bar{x}_3\bar{x}_2\bar{x}_1 \cup \bar{x}_3\bar{x}_2x_1 \cup \bar{x}_3x_2x_1,$$

Scanned with OKEN Scanner

adică

$$D_{2120}^{x_1x_2x_3} = D_{320}^{x_3x_2x_1}.$$

Funcţiile logice de n variabile de intrare vor fi descrise de simbolurile de marcare cuprinse între

$$D_0^{x_1x_2\ldots x_n} \quad \text{şi} \quad D_{(2^{2^n}-1-1)_8}^{x_1x_2\ldots x_n}.$$

O parte dintre aceste funcţii logice sînt reductibile.

Dintre funcţiile logice ireductibile, funcţiile logice care au simboluri de marcare cuprinse între

$$D_{(2^{2^n}-1)_8}^{x_1x_2\cdots x_n} \quad \text{şi} \quad D_{(2^{2^n}-1)_8}^{x_1x_2x_3\cdots x_n}$$

reprezintă complementele funcţiilor logice cu simboluri de marcare cuprinse între

$$D_0^{x_1x_2\ldots x_n} \quad \text{şi} \quad D_{(2^{2^n-1}-1)_8}^{x_1x_2\ldots x_n}.$$

Această concluzie rezultă ca o consecinţă directă a proprietăţii de complementaritate a simbolului de marcare, proprietate exprimată prin formula (3.24).

Într-adevăr, dacă prima jumătate a funcţiilor logice posibile de n variabile de intrare cuprinde numerele de stare de la 0 la $(2^{2^n}-1-1)_8$, care reprezintă jumătate din totalul numerelor de stare posibile, prin negarea logică a acestora se obţine cealaltă jumătate a numerelor de stare, adică cele cuprinse între $(2^{2^n}-1)_8$ şi $(2^{2^n}-1-1)_8$. Dar prin această negare logică se obţin funcţiile logice care corespund numerelor de ordine negate, adică, potrivit formulei (3.24), complementelor primelor funcţii logice.

De asemenea, după cum s-a arătat în paragraful 3.4, există un număr de funcţii logice care pot fi obţinute prin permutări ale indicilor simbolurilor de marcare ale primei jumătăţi din numărul

de funcții posibile de n variabile. Pe calea permutării indicilor se pot găsi simboluri de marcare diferite ce corespund aceleiași funcții logice.

Funcțiile logice de numai o variabilă binară și cele de două variabile binare sînt considerate ca definitorii în algebra booleeană, adică ca funcții ce definesc opreațiile logice de bază în această algebră.

Considerațiile de mai sus permit selectarea acelor funcții logice de o variabilă binară și de două variabile binare care pot servi ca punct de plecare în deducerea tuturor funcțiilor logice caracteristice funcțiilor de o variabilă binară și de două variabile binare. Vom numi aceste funcții logice „esențiale". În literatura de specialitate, sub numele de „funcții logice esențiale" se cuprind cele patru funcții de o variabilă binară și 8 funcții de două variabile binare.

Avînd în vedere posibilitățile oferite de simbolul de marcare putem să ne limităm la numai două funcții de o variabilă binară și la numai 4 funcții de două variabile binare. Pentru o singură variabilă binară neexistînd posibilități de permutare sau de reducere, funcțiile logice esențiale vor corespunde numerelor de ordine cuprinse între

$$N = 0 \quad \text{și} \quad N = (2^{2^1} - 1 - 1)_8 = 1.$$

Acestea vor fi

$$D_0^{x_0} = 0 \qquad \text{(constanta 0)}$$

și

$$D_1^{x_0} = D_1^{x_0} = x_0 \qquad \text{(variabila binară)}.$$

Restul celorlalte funcții logice, care au numere de ordine cuprinse între $N = (2^{2^1} - 1)_8 = 2$ și $N = (2^{2^1} - 1)_8 = 3$, vor reprezenta complementele primelor funcții.

Într-adevăr, avem

$$D_2^{x_0} = \overline{D}_2^{x_0} = \overline{D}_1^{x_0} = \overline{x}_0 \qquad \text{(negația logică)}$$

şi

$$D_3^{x_0} = \overline{D}_{\overline{3}}^{x_0} = \overline{D}_0^{x_0} = 1 \qquad \text{(constanta 1).}$$

Pentru două variabile binare, funcţiile logice esenţiale se vor găsi printre funcţiile logice ale căror numere de ordine sînt cuprinse între

$$N = 0 \quad \text{şi} \quad N = (2^{2^2} - 1 - 1)_8 = 7.$$

Dintre acestea, funcţiile logice exprimate prin simbolurile de marcare

$$D_0^{x_0x_1} = 0,$$

$$D_3^{x_0x_1} = D_{(2,\,3)}^{x_0x_1} = D_{(1)}^{x_0} = x_0,$$

$$D_5^{x_0x_1} = D_{(1,\,3)}^{x_0x_1} = D_{(1)}^{x_1} = x_1,$$

sînt funcţii logice reductibile la funcţii de o singură variabilă binară.

De asemenea funcţia logică corespunzătoare la $N = 4$ se poate obţine prin permutarea variabilelor conţinute de funcţia logică pentru $N = 2$.

Într-adevăr,

$$D_4^{x_0x_1} = D_{(1)}^{x_0x_1} = \overline{x}_0 x_1$$

şi

$$x_0x_1 = D_{(2)}^{x_0x_1} = D_2^{x_0x_1}.$$

Rămîn deci ca funcţii logice esenţiale de două variabile binare funcţiile date de simbolurile de marcare:

$$D_1^{x_0x_1} = D_{(3)}^{x_0x_1} = x_0 x_1 \qquad \text{(conjuncţia logică),}$$

$$D_2^{x_0x_1} = D_{(2)}^{x_0x_1} = x_0\bar{x}_1 \qquad \text{(interdicția logică),}$$

$$D_6^{x_0x_1} = D_{(1,\,2)}^{x_0x_1} = \bar{x}_0x_1 \cup x_0\bar{x}_1 \qquad \text{(suma modulo 2, numită și „sau exclusiv''),}$$

$$= x_0 \oplus x_1,$$

$$D_7^{x_0x_1} = D_{(1,\,2,\,3)}^{x_0x_1} = x_0 \cup x_1 \qquad \text{(disjuncția logică).}$$

Complementele funcțiilor logice esențiale vor fi date de

$$\bar{D}_1^{x_0x_1} = D_{\bar{1}}^{x_0x_1} = D_{16}^{x_0x_1} = D_{(0,\,1,\,2)}^{x_0x_1} = \bar{x}_0 \cup \bar{x}_1 =$$

$$= \overline{x_0x_1} = x_0 \perp x_1 \qquad \text{(funcția lui Sheffer „numai''),}$$

$$\bar{D}_2^{x_0x_1} = D_{\bar{2}}^{x_0x_1} = D_{15}^{x_0x_1} = D_{(0,\,1,\,3)}^{x_0x_1} =$$

$$= \bar{x}_0 \cup x_1 = \overline{x_0\,\bar{x}_1} = x_0 \to x_1 \qquad \text{(implicația logică),}$$

$$\bar{D}_6^{x_0x_1} = D_{\bar{6}}^{x_0x_1} = D_{11}^{x_0x_1} = D_{(0,\,3)}^{x_0x_1} =$$

$$= \bar{x}_0\bar{x}_1 \cup x_0x_1 = \overline{x_0 \oplus x_1} = x_0 \sim x_1 \qquad \text{(echivalența logică),}$$

$$\bar{D}_7^{x_0x_1} = D_{\bar{7}}^{x_0x_1} = D_{10}^{x_0x_1} = D_{(0)}^{x_0x_1} =$$

$$= \bar{x}_0\bar{x}_1 = \overline{x_0 \cup x_1} = x_0 \top x_1 \qquad \text{(funcția lui Peirce „nici'').}$$

Atît funcțiile logice esențiale, cît și complementele lor au primit în literatură denumiri specifice. Acestea sînt scrise mai sus alături de funcțiile logice pe care le definesc.

Scanned with OKEN Scanner

Capitolul 6

METODĂ SIMBOLICĂ DE MINIMIZARE A EXPRESIILOR BOOLEENE

După cum s-a arătat în capitolele anterioare ale acestei lucrări, din tabela de adevăr sau din simbolul de marcare se poate deduce forma canonică disjunctivă sau conjunctivă a funcţiei logice care corespunde unei scheme de comutaţie.

Realizarea practică a schemei de comutaţie după forma canonică a expresiei logice este însă neeconomică, necesitînd, de regulă, un număr mai mare de contacte de relee sau de module logice (elemente de comutaţie statică) decît cel strict necesar îndeplinirii funcţiei logice cerute. Se impune deci aducerea expresiei logice la o formă minimală, adică la o formă care să conţină numărul strict necesar de termeni şi de variabile conţinute de aceştia.

Pînă acum au fost propuse mai multe grupe de metode folosite pentru minimizarea expresiilor booleene, dintre care mai importante sînt : metoda transformărilor algebrice, metodele algebrice de tipul celei a lui Quine şi Mc Cluskey, metode matriceale şi grupul de metode grafice care folosesc matrici de combinaţii, cum este metoda diagramelor Karnaugh sau Veitch.

Toate aceste metode de minimizare, expuse în majoritatea lucrărilor de specialitate, necesită fie rezolvări matematice relativ complicate, fie o serie de construcţii grafice, cum ar fi matricele de combinaţii şi cele de implicanţi primi, care devin laborioase mai ales atunci cînd numărul de variabile devine mare.

Metodele de minimizare amintite folosesc adiacenţele dintre diferitele combinaţii ale variabilelor binare şi legile de absorbţie şi de incluziune ale algebrei booleene în scopul minimizării expresiei logice a schemei de comutaţie.

Metoda simbolică de minimizare, care se expune în cele ce urmează, este o metodă de tipul celei a lui Quine şi McCluskey. Ea diferă de aceasta prin modul de găsire a grupurilor adiacente şi se aseamănă cu această metodă prin faptul că foloseşte combinaţii octale de tipul numerelor de stare.

Scanned with OKEN Scanner

Proprietățile simbolului de marcare descrise de formulele (3.8) și (3.9) pot fi folosite în scopul formării de grupe de numere de stare adiacente.

În cazul cînd aceste grupe, cuprinse în simboluri de marcare distincte, reunesc numărul maxim posibil de numere de stare adiacente dintre cele conținute de simbolul inițial, formula (3.8) reprezintă forma iredondantă a expresiei booleene definite de acest simbol. Ea este numită adesea formă minimală avînd în vedere că foarte frecvent ea coincide cu această formă. Forma minimală depinde însă de criteriul de minim luat în considerare. În acest caz, presupunînd că fiecare număr de stare din simbolul inițial care se descompune apare o dată cel puțin într-o astfel de grupare, fiecare termen al formulei (3.8) reprezintă un implicant prim.

Metoda folosită pentru formarea grupelor de numere de stare adiacente poate fi dedusă din regulile de simetrie folosite la construcția grafului aferent unui simbol de marcare. În loc de a construi graful însă, construcție care cere timp, noi ne vom mulțumi cu întocmirea unei liste de numere de stare adiacente. Această listă va cuprinde numerele de stare aferente nodurilor grafului, evidențiind diferențele de adiacență care caracterizează ramurile lui.

Construcția grafului începe prin figurarea primului nod, care corespunde unuia dintre numerele de stare reprezentat în cod binar printr-un număr minim de biți 1. Presupunînd că numerele de stare cuprinse de indicele inferior al simbolului de marcare sînt scrise în ordinea lor crescătoare, primul nod va fi reprezentat de primul număr de stare din șirul conținut de acest indice.

Întocmirea „listei de adiacențe" asociate grafului va începe de asemenea prin scrierea acestui număr de stare cap de listă. Construcția grafului se continuă apoi prin figurarea unui nod adiacent primului și trasarea primei ramuri a lui. Pe ramura grafului se înscrie fie numărul diferență de adiacență, fie variabila care dispare prin disjuncția combinațiilor reprezentate de cele două noduri care au servit la trasarea ei. Evident, graful conține o astfel de ramură numai dacă printre numerele de stare conținute de simbolul de marcare se găsește cel puțin un număr de stare adiacent primului înscris pînă acum. Dacă însă există mai multe astfel de numere de stare, se consideră numărul de stare cel mai apropiat ca valoare de primul număr de stare folosit deja. Se recomandă sondarea adiacențelor în ordinea crescătoare a diferențelor de adiacență posibile.

Dacă indicele inferior al simbolului de marcare nu conţine nici un număr de stare adiacent primului, aceasta înseamnă că primul implicant prim va cuprinde numai numărul de stare aferent nodului singular al grafului.

Întocmirea listei de adiacenţe aferente grafului se continuă prin înscrierea numărului de stare adiacent primului, imediat după o bară verticală notată prin numărul care reprezintă diferenţa de adiacenţă dintre cele două numere de stare. Cele două numere de stare folosite se bifează în şirul numerelor de stare cuprinse de simbolul de marcare. Bifarea se face numai cu scopul de a ţine cont de faptul că numerele de stare bifate au folosit cel puţin o dată la formarea grafului sau întocmirea listei de adiacenţe.

A doua ramură a grafului, simetrică primei, va exista dacă simbolul de marcare conţine alte două numere de stare care corespund aceleiaşi diferenţe de adiacenţă în raport cu primele două numere de stare. În caz afirmativ se construieşte patrulaterul simetric avînd drept colţuri cele patru numere de stare adiacente între ele.

În lista de adiacenţe, următoarele două numere de stare adiacente primelor se scriu imediat după bara verticală, notată prin numărul diferenţă de adiacenţă care satisface adiacenţa cerută şi este deci luat în considerare. Aceste numere se bifează de asemenea în şirul numerelor de stare conţinute de simbolul de marcare.

Se sondează apoi dacă şirul numerelor de stare cuprinde alte patru numere de stare adicente primelor, care să corespundă unui alt număr diferenţă de adiacenţă, dar comun celor patru numere de stare. În caz afirmativ se figurează al doilea patrulater simetric al grafului, ale cărui colţuri se unesc cu colţurile primului patrulater, formînd astfel în ansamblu un hexaedru simetric.

În lista de adiacenţe, următoarele patru numere de stare se trec imediat după bara verticală notată prin numărul diferenţă de adiacenţă luat în considerare, bifîndu-se aceste numere în şirul numerelor de stare conţinute de simbolul de marcare.

Construcţia grafului se continuă la fel, căutînd următoarele opt numere de stare adiacente între ele şi adiacente toate cu primele opt numere de stare care au servit la figurarea hexaedrului simetric. Se procedează astfel pînă se epuizează toate posibilităţile de grupare.

La fel se continuă şi scrierea listei de adiacenţe, scriind de fiecare dată după bara notată prin numărul diferenţă de adiacenţă

Scanned with OKEN Scanner

grupul de numere de stare care corespund aceleiaşi diferenţe de adiacenţă în raport cu cele înscrise pînă atunci în listă.

Dacă spaţiul nu mai permite înscrierea numerelor de stare ale listei de adiacenţe pe orizontală, aceasta se continuă pe verticală, aşa cum se arată în exemplele ce urmează. În cazul continuării listei de adiacenţe pe verticală, nu este necesar să scriem toate numerele de stare care completează liniile orizontale ci numai numerele de stare cap de linie, verificînd însă existenţa acestora în şirul numerelor de stare conţinute de simbolul de marcare şi bifîndu-le. Verificarea se referă la existenţa în simbol a tuturor numerelor de stare cerute pentru formarea grupării conform regulii date mai sus.

Pentru explicitarea implicantului prim corespunzător listei de adicenţe întocmite se scrie simbolul de marcare, trecînd în indicele superior şirul variabilelor de intrare, iar în indicele inferior expresia binară a numărului cap de listă. Se taie apoi (prin subliniere) variabilele binare şi biţii reprezentativi ai numerelor diferenţă de adiacenţă notate în dreptul barelor verticale sau orizontale (cînd lista este continuată pe verticală) din lista stărilor adiacente. Simbolul de marcare rezultat corespunde implicantului prim căutat.

Procedeul aplicat pentru calculul primului implicant prim se aplică la fel şi pentru calculul celorlalţi implicanţi primi. Drept cap de listă se ia de fiecare dată primul număr de stare din simbolul de marcare rămas nebifat în şirul numerelor de stare corespunzătoare valorii binare 1 a variabilei de ieşire.

Se continuă astfel formarea de grupări pînă sînt epuizate toate numerele de stare specificate de simbolul de marcare pentru valoarea 1 a variabilei de ieşire.

După cum s-a văzut mai sus, regulile după care se construieşte graful asociat unui simbol de marcare ne-a condus la stabilitatea regulilor după care se poate întocmi lista numerelor de stare adiacente. Rezultă deci că pentru găsirea implicaţilor primi ai unei expresii booleene ne putem lipsi de construcţia grafului, întocmind numai lista de adiacenţe după regulile stabilite mai sus.

Exemplul 6.1. Să se găsească implicanţii primi ai simbolului de marcare

$$F(A)=D^{abcde}_{(0,1,2,3,4,5,6,7,12,13,16,17,22,23,26,27,32,33,36,37)}.$$

Pentru comparaţie, alături de lista de adiacenţe se va construi şi graful aferent simbolului de marcare dat. Folosind regulile date mai sus, se construieşte graful aferent simbolului de marcare din figura 6.1. şi se întocmeşte lista de adiacenţe dată mai jos :

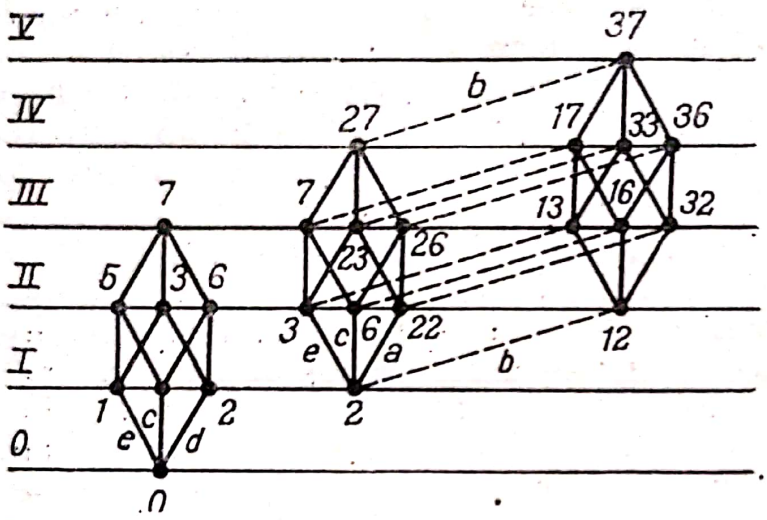


Fig. 6.1. — Graful aferent exemplului 6.1.

implicantul 1 :

$$4\,\frac{0}{4}\;/\;\overset{1}{1}\;/\;\overset{2}{2}\;3 \qquad 5\;\;6\;\;7 \qquad D^{abcde}_{\underline{00000}} = \bar{a}\bar{b}.$$

Numerele de stare (5), (6) şi (7) nu era necesar să fie trecute în listă, verificînd însă existenţa lor şi bifîndu-le în şirul numerelor de stare conţinute de simbolul de marcare.
În cazul nostru, ele au fost trecute numai pentru a arăta cum se găsesc aceste numere;

implicantul 2 :

$$20\,\frac{12}{32}\;/\;\overset{1}{13}\;/\;\overset{4}{16}\;\overset{10}{17}\;/\;2\;3\;6\;7 \qquad D^{abcde}_{\underline{01010}} = d.$$

Rezultă deci următoarea expresie booleeană minimală pentru simbolul de marcare dat :

$$F(A) = \bar{a}\bar{b} \cup d.$$

Exemplul 6.2. Se cere minimizarea expresiei booleene prin metoda simbolică pentru funcția logică dată prin simbolul de marcare

$$F(X) = D^{ABCD}_{(0,\ 2,\ 5,\ 6,\ 7,\ 10,\ 11,\ 13,\ 14\ 15,\ 16,\ 17)}.$$

Aplicînd regulile metodei simbolice de minimizare expuse mai sus, se găsesc succesiv implicanții primi

— *implicantul 1*

$$\begin{array}{c} \overset{2}{} \\ 0\ /\ 2 \end{array} \qquad D^{ABCD}_{00\underline{0}\,0} = \overline{A}\,\overline{B}\,\overline{D};$$

— *implicantul 2*

$$\begin{array}{c} \overset{2}{}\quad\overset{10}{} \\ 5\ /\ 7\ /\ 15\ 17 \end{array} \qquad D^{ABCD}_{\underline{0}1\underline{0}1} = BD;$$

— *implicantul 3*

$$\begin{array}{c} \overset{1}{}\quad\overset{10}{} \\ 6\ /\ 7\ /\ 16\ 17 \end{array} \qquad D^{ABCD}_{\underline{0}11\underline{0}} = BC;$$

— *implicantul 4*

$$\begin{array}{c} \overset{1}{}\quad\overset{4}{} \\ 10\ /\ 11\ /\ 14\ 15 \end{array} \qquad D^{ABCD}_{1\underline{0}0\underline{0}} = A\overline{C};$$

— *implicantul 5*

$$\begin{array}{c} \overset{2}{}\quad\overset{4}{} \\ 13\ /\ 11\ /\ 17\ 15 \end{array} \qquad D^{ABCD}_{1\underline{0}1\,1} = AD.$$

Rezultă deci final expresia minimală

$$F(X) = \overline{A}\,\overline{B}\,\overline{D} \cup BD \cup BC \cup A\overline{C} \cup AD.$$

Exemplul acesta este rezolvat prin metoda matriceală în [25, p. 44—46].

Prin comparație rezultă că metoda simbolică de minimizare conduce la același rezultat, dar pe o cale mai expeditivă decît cea folosită în lucrarea citată.

În cazul cînd simbolul de marcare conține și numere de stare indiferente, folosirea acestora în lista de adiacențe nu este obli-

gatorie, întrucît fie că combinaţiile respective ale variabilelor binare nu apar în funcţionarea schemei de comutaţie descrise de simbol, fie că valoarea variabilei de ieşire poate fi 0 sau 1, fără a influenţa condiţiile de funcţionare cerute acestei scheme.

Numerele de stare indiferente vor fi folosite deci numai în măsura în care ele contribuie la formarea unor grupări mai mari de numere de stare (noduri) adiacente.

Modul de folosire a numerelor de stare indiferente în cazul cînd se alege calea de minimizare prin gruparea nodurilor 1 poate diferi de cel folosit în cazul cînd se alege calea de minimizare prin gruparea nodurilor 0. Cele două căi posibile de minimizare pot deci conduce la rezultate diferite, de aceea în astfel de cazuri este recomandabilă rezolvarea problemei pe ambele căi şi alegerea rezultatului optim din punctul de vedere al realizării practice a schemei de comutaţie.

Facem de asemenea remarca că în cazul inexistenţei stărilor indiferente cele două căi de minimizare conduc la formule logice complementare. Această concluzie rezultă şi din formulele (3.24) şi (3.25). Astfel, în cazul folosirii grupărilor de noduri 0, se obţine funcţia logică negată logic. Complementul acesteia va reprezenta funcţia logică căutată.

Exemplul 6.3. Se cere minimizarea expresiei booleene date prin simbolul de marcare

$$F(Z) = D^{abcd}_{1^*,\,3,\,4^*,\,5,\,7^*,\,10,\,12^*,\,13,\,14,\,15,\,17)}.$$

Metoda prin grupe de „1", ne conduce la stabilirea următorilor implicanţi primi :

— *implicantul 1*

$$\begin{array}{cccc} & 4 & & 10 \\ 3 & / & 7^* & / \quad 13 \quad 17 \end{array} \qquad D^{abcd}_{\underline{00}11} = cd;$$

— *implicantul 2*

$$\begin{array}{cccc} 1 & & 10 & \\ 5 & / & 4^* & / \quad 15 \quad 14 \end{array} \qquad D^{abcd}_{0\underline{10}1} = b\bar{c};$$

— *implicantul 3*

$$\begin{array}{ccc} & 4 & \\ 10 & / & 14 \end{array} \qquad D^{abcd}_{1\underline{00}0} = a\bar{c}\,\bar{d}.$$

Rezultă deci

$$F(Z) = cd \cup b\bar{c} \cup a\bar{c}\bar{d}.$$

Luarea în considerare a nodurilor 0, potrivit formulei (3.25), se face prin negarea numărului de ordine al funcției logice date de simbolul de marcare. Astfel putem scrie

$$F(Z) = \overline{D}\frac{abcd}{\overline{N}}.$$

Numerele de stare corespunzătoare lui $\overline{N}$ caracterizează valorile $F(Z) = 0$ și sînt de fapt cele necuprinse de simbolul de marcare dat prin enunțul problemei, astfel încît putem scrie imediat

$$F(Z) = \overline{D}^{abcd}_{(0,\ 1^*,\ 2,\ 4^*,\ 6,\ 7^*,\ 11,\ 12^*,\ 16)}.$$

Folosind acest simbol de marcare, deducem la fel ca mai sus următorii implicanți primi ai funcției logice $\overline{F}(Z)$:

$$\begin{matrix} 2 & 4 \\ 0 & / & 2 & / & 4^* & 6 \end{matrix} \qquad D^{abcd}_{0\underline{0}\underline{0}0} = \bar{a}\bar{d},$$

$$\begin{matrix} & 10 \\ 11 & / & 1^* \end{matrix} \qquad D^{abcd}_{\underline{1}001} = \bar{b}\bar{c}d,$$

$$\begin{matrix} 4 & & 10 \\ 16 & / & 12^* & / & 6 & 2 \end{matrix} \qquad D^{abcd}_{\underline{11}10} = c\bar{d}.$$

Rezultă deci

$$\overline{F}(Z) = \bar{a}\bar{d} \cup \bar{b}\bar{c}d \cup c\bar{d}.$$

Observăm că

$$F(Z) = \overline{\bar{a}\bar{d} \cup \bar{b}\bar{c}d \cup c\bar{d}} = (a \cup d)(b \cup c \cup \bar{d})(\bar{c} \cup d) =$$
$$= a\bar{c}\bar{d} \cup acd \cup bd \cup ab\bar{c},$$

deci rezultatele obținute pe cele două căi de minimizare nu sînt complementare. Acest rezultat apare ca urmare a folosirii diferite a numerelor de stare indiferente în cele două căi de minimizare.

Acest exemplu este rezolvat prin altă metodă în [56, p. 52].

Exemplul 6.4. Să se minimizeze expresia booleeană definită prin simbolul de marcare

$$F(X) = D^{ABC}_{(0,\,1,\,2,\,3,\,4,\,7)}.$$

Pentru comparaţie se poate vedea cum este rezolvată această problemă prin metoda Quine McCluskey, rezolvare dată în [72, p. 93—94].

Folosind metoda simbolică de minimizare, obţinem imediat implicanţii primi

$$\begin{array}{l} 1\ \ 2 \\ 0\ /\ 1\ /\ 2 \end{array} \qquad D^{ABC}_{0\,\underline{0\,0}} = \overline{A},$$

$$\begin{array}{l} 4 \\ 4\ /\ 0 \end{array} \qquad D^{ABC}_{\underline{1}\,0\,0} = \overline{B}\,\overline{C},$$

$$\begin{array}{l} 4 \\ 7\ /\ 3 \end{array} \qquad D^{ABC}_{\underline{1}\,1\,1} = BC.$$

Rezultă deci expresia logică minimală

$$F(X) = \overline{A} \cup \overline{B}\,\overline{C} \cup BC.$$

Acelaşi rezultat se obţine şi prin metoda Quine McCluskey dar pe o cale mai laborioasă.

Exemplul 6.5. Pentru a oferi un mijloc de comparaţie cu metoda Quine McCluskey, ne propunem să găsim forma minimală a expresiei booleene date de

$$F(A) = D^{ABCD}_{(0,\,1,\,2,\,5,\,6,\,7,\,11,\,12,\,13,\,15,\,17)}.$$

Prin metoda simbolică de minimizare se obţin implicanţii primi

$$\begin{array}{l} 1 \\ 0\ /\ 1 \end{array} \qquad D^{ABCD}_{0\,0\,0\,\underline{0}} = \overline{A}\,\overline{B}\,\overline{C},$$

$$\begin{array}{l} 4 \\ 2\ /\ 6 \end{array} \qquad D^{ABCD}_{0\,\underline{0}\,1\,0} = \overline{A}\,C\,\overline{D},$$

$$\begin{array}{l} 2\ \ 10 \\ 5\ /\ 7\ /\ 15\ 17 \end{array} \qquad D^{ABCD}_{\underline{0}\,1\,\underline{0}\,1} = BD,$$

$$\begin{array}{l} \overset{2}{11} \,/\, \overset{4}{13} \,/\, 15\ 17 \end{array} \qquad D\begin{matrix} ABCD \\ 1\underline{00}1 \end{matrix} = AD,$$

$$\begin{array}{l} \overset{1}{12} \,/\, 13 \end{array} \qquad D\begin{matrix} ABCD \\ 101\underline{0} \end{matrix} = A\bar{B}C.$$

Rezultă deci

$$F(A) = \bar{A}B\bar{C} \cup \bar{A}C\bar{D} \cup BD \cup AD \cup A\bar{B}C.$$

Acest exemplu este tratat prin metoda Quine McCluskey în lucrarea [72, p. 95].

Exemplul 6.6. Pentru a oferi un mijloc de comparaţie cu metoda incluziunii iterative a lui Tison [72, p. 96—99], ne propunem minimizarea prin metoda simbolică a expresiei booleene:

$$F = a\bar{b}\bar{d} \cup abde \cup ab\bar{e} \cup ac\bar{d} \cup b\bar{c}\bar{d} \cup \bar{b}\bar{d}.$$

Se impune mai întîi aducerea acestei formule la forma canonică disjunctivă. Acest lucru se poate face expeditiv folosind metoda simbolică de analiză, expusă în capitolul 7 al acestei lucrări. Aici vom folosi însă metoda obişnuită a transformărilor algebrice. Se obţine astfel succesiv

$$F = a\bar{b}c\bar{d} \cup a\bar{b}\bar{c}\bar{d} \cup abcde \cup ab\bar{c}de \cup abc\bar{e} \cup ab\bar{c}\bar{e} \cup abc\bar{d} \cup$$

$$\cup a\bar{b}c\bar{d} \cup ab\bar{c}\bar{d} \cup \bar{a}b\bar{c}\bar{d} \cup a\bar{b}\bar{d} \cup \bar{a}\bar{b}\bar{d} =$$

$$= a\bar{b}cde \cup a\bar{b}cd\bar{e} \cup a\bar{b}\bar{c}de \cup a\bar{b}\bar{c}d\bar{e} \cup abcde \cup ab\bar{c}de \cup,$$

$$\cup abcd\bar{e} \cup abc\bar{d}\bar{e} \cup ab\bar{c}d\bar{e} \cup ab\bar{c}d\bar{e} \cup abc\bar{d}e \cup abc\bar{d}\bar{e} \cup,$$

$$\cup al\bar{c}de \cup ab\bar{c}d\bar{e} \cup \bar{a}b\bar{c}de \cup \bar{a}b\bar{c}d\bar{e} \cup a\bar{b}cde \cup a\bar{b}cd\bar{e} \cup,$$

$$\cup a\bar{b}\bar{c}de \cup a\bar{b}\bar{c}d\bar{e} \cup a\bar{b}cde \cup a\bar{b}cd\bar{e} \cup \bar{a}\bar{b}\bar{c}de \cup \bar{a}\bar{b}\bar{c}d\bar{e} \cup \bar{a}\bar{b}cde \cup \bar{a}\bar{b}cd\bar{e}.$$

Scanned with OKEN Scanner

Rezultă simbolul de marcare

$$F = D^{abcde}_{(0,\,1,\,4,\,5,\,10,\,11,\,20,\,21,\,24,\,25,\,30,\,31,\,32,\,33,\,34,\,35,\,36,\,37)}.$$

Implicanţii primi corespunzători vor fi daţi de

$$\begin{array}{l} \overset{1}{0} \,/\, \overset{4}{1} \,/\, \overset{\quad 20}{4\ 5} \,/\, 20\ 21\ 24\ 25 \end{array} \qquad D^{abcde}_{\underline{0}0\underline{0}\underline{0}0} = \bar{b}\bar{d},$$

$$\begin{array}{l} \overset{1}{10} \,/\, \overset{10}{11} \,/\, \overset{\quad 20}{0\ 1} \,/\, 30\ 31\ 20\ 21 \end{array} \qquad D^{abcde}_{\underline{01}000} = \bar{c}\bar{d},$$

$$\begin{array}{l} \overset{1}{32} \,/\, \overset{2}{33} \,/\, \overset{\quad 4}{30\ 31} \,/\, 36\ 37\ 34\ 35 \end{array} \qquad D^{abcde}_{11\underline{010}} = ab.$$

Rezultă deci

$$F = \bar{b}\bar{d} \cup \bar{c}\bar{d} \cup ab.$$

Exemplul 6.7. Se cere stabilirea expresiei logice minimale care corespunde simbolului de marcare

$$F(Z) = D^{ABCDEabcde}_{\begin{array}{l} 1:(0.164\ 0.262\ 0.264\ 0.321\ 0.322\ 0.324\ 0.454\ 0.461 \\ \quad 0.464\ 0.512\ 0.514\ 0.521\ 0.522\ 0.524\ 0.611\ 0.612 \\ \quad 0.614\ 0.622\ 0.624\ 1.046\ 1.051\ 1.052\ 1.054\ 1.061 \\ \quad 1.062\ 1.064\ 1.105\ 1.106\ 1.111\ 1.112\ 1.114\ 1.121 \\ \quad 1.122\ 1.124\ 1.203\ 1.205\ 1.206\ 1.211\ 1.212\ 1.214 \\ \quad 1.221\ 1.222\ 1.224), \\ 0:(0.143\ 0.145\ 0.146\ 0.151\ 0.152\ 0.154\ 0.161\ 0.162 \\ \quad 0.170\ 0.243\ 0.245\ 0.246\ 0.251\ 0.252\ 0.254\ 0.261 \\ \quad 0.270\ 0.303\ 0.305\ 0.306\ 0.311\ 0.312\ 0.314\ 0.330 \\ \quad 0.443\ 0.445\ 0.446\ 0.451\ 0.452\ 0.470\ 0.503\ 0.505 \\ \quad 0.506\ 0.511\ 0.530\ 0.603\ 0.605\ 0.606\ 0.630\ 1.043 \\ \quad 1.045\ 1.070\ 1.103\ 1.130\ 1.230\ 1.403\ 1.405\ 1.406 \\ \quad 1.411\ 1.412\ 1.414\ 1.421\ 1.422\ 1.424\ 1.430). \end{array}}$$

Numerele de stare necuprinse mai sus sînt indiferente ca metodică generală în cazul unor probleme cu multe nedeterminate ca cea din exemplul nostru, se recomandă sondarea diferenţelor de

adiacență în ordinea lor crescătoare. În acest fel este mult ușurată procedura de determinare a numerelor de stare.

În listele de adiacențe date mai jos s-au scris numai numerele de stare ale primei linii, care cuprinde numai prima grupă de opt numere de stare ce satisfac condițiile de adiacență impuse de metoda folosită.

Din grupa următoare de numere de stare, care completează cea de a doua linie, s-a scris numai numărul de stare cap de linie, verificînd însă existența celorlalte numere de stare ale liniei printre numerele de stare date prin simbolul de marcare la valoarea 1 a variabilei Z, sau printre numerele de stare indiferente. Este suficient ca un singur număr de stare să nu fie găsit astfel pentru ca diferența de adiacență să nu fie luată în considerare. Astfel, la stabilirea listei pentru implicantul prim 1 n-a fost luată în considerare, diferența de adiacență 4, întrucît numărul de stare 0.161 se găsea printre numerele de stare interzise, date în simbolul de marcare pentru $Z = 0$.

Din grupa a treia de numere de stare, care completează următoarele două linii, s-au scris de asemenea numai cele două numere de stare cap de linie, procedînd apoi la fel ca mai sus.

Sondînd diferențele de adiacență în ordinea lor crescătoare, se completează lista pînă se epuizează toate numerele diferență de adiacență, în cazul exemplului nostru, numerele: 1, 2, 4, 10, 20, 40, 100, 200, 400, și 1000.

Se procedează apoi, așa cum s-a arătat la alte exemple, la întocmirea următoarei liste de adiacență, care va corespunde următorului implicant prim. Calculul se termină în momentul cînd în aceste liste, am cuprins, o dată cel puțin, fiecare număr de stare dat de simbolul de marcare pentru $Z = 1$ din lista de adiacențe.

Dacă pe prima linie orizontală apar numai diferențele de adiacență 1,2 și 4, atunci celelalte linii orizontale vor conține numere de stare care diferă față de numerele cap de linie prin cifra semnificativă a unităților (0,....,7), astfel încît verificarea făcîndu-se foarte ușor, nu mai este necesar să înscriem aceste numere în listă ci numai să verificăm existența lor în șirul numerelor de stare corespunzătoare valorii binare 1 a variabilei de ieșire și inexistența acestora în șirul numerelor de stare corespunzătoare valorii binare 0 a acestei variabile. Se face astfel economie de timp și de hîrtie.

Scanned with OKEN Scanner

În cazul altor diferenţe de adiacenţă pe prima linie se poate proceda la fel ca mai sus, după ce s-a dobîndit o rutină în acest sens.

În cazul nostru, folosind metoda simbolică de minimizare, obţinem succesiv :

— *implicantul 1*

	1	2	10				
40	0.164	/ 0.165	/ 0.166	0.167	/ 0.174	0.175	0.176	0.177
	0.124						
100	0.064						
	0.024	400					
200	0.364		0.564				
	0.324		0.524				
	0.264		0.464				
	0.224		0.424				
			0.764				
			0.724				
			0.664				
			0.624				

$$D^{ABCDEabcde}_{00011\underline{1}0\underline{1}\underline{0}\underline{0}} = \bar{A}ac;$$

— *implicantul 2*

	1	4	10				
40	0.262	/ 0.263	/ 0.266	0.267	/ 0.272	0.273	0.276	0.277
	0.222						
100	0.362						
	0.322						
400	0.662						
	0.622						
	0.762						
	0.722						
1.000	1.262						
	. . .						

$$D^{ABCDEabcde}_{\underline{0}\underline{0}1\underline{0}\underline{1}1\underline{0}\underline{0}1\underline{0}} = Cad;$$

— *implicantul 3*

$$\begin{array}{rl} & \overset{1}{0.321} \,/\, 0.320 \,/\, 0.323 \quad \overset{2}{0.322} \,/\, 0.325 \quad 0.324 \quad 0.327 \quad 0.326 \\ 40 & \overline{0.361} \\ 400 & \overline{0.721} \\ 1000 & \overline{0.761} \\ & \overline{1.321} \\ & \ldots \end{array} \qquad D^{ABCDEbacde}_{\underline{00}11\underline{0}10\underline{001}} = CDa\bar{b};$$

— *implicantul 4*

$$\begin{array}{rl} & \overset{1}{0.454} \,/\, 0.455 \,/\, 0.456 \quad \overset{20}{0.457} \,/\, 0.474 \quad 0.475 \quad 0.476 \quad 0.477 \\ 40 & \overline{0.414} \\ 100 & \overline{0.554} \\ 200 & \overline{0.654} \\ & \ldots \end{array} \qquad D^{ABCDEabcde}_{01\ \underline{001}\,\underline{0}11\underline{00}} = \bar{A}Bbc;$$

— *implicantul 5*

$$\begin{array}{rl} & \overset{1}{0.461} \,/\, 0.460 \,/\, 0.463 \quad \overset{4}{0.462} \,/\, 0.465 \quad 0.464 \quad 0.467 \quad 0.466 \\ \underline{4}0 & \overline{0.421} \\ 100 & \overline{0.561} \\ 200 & \overline{0.661} \end{array} \qquad D^{ABCDEabcde}_{01\underline{0}\,\underline{01}\,10\underline{001}} = \bar{A}Ba\bar{b};$$

— *implicantul 6*

$$\begin{array}{rl} & \overset{1}{0.512} \,/\, 0.513 \,/\, 0.516 \quad \overset{20}{0.517} \,/\, 0.532 \quad 0.533 \quad 0.536 \quad 0.537 \\ 40 & \overline{0.552} \\ 200 & \overline{0.712} \\ 1000 & \overline{0.752} \\ & \overline{1.512} \\ & \ldots \end{array} \qquad D^{ABCDEabcde}_{\underline{0}1\underline{0}1\underline{00}\,\underline{0}1\underline{0}1} = BDbd;$$

— *implicantul 7*

$$\begin{array}{rl} & \overset{1}{} \quad \overset{2}{} \quad \overset{4}{} \\ 40 & \underline{0.611} \,/\, 0.610 \,/\, 0.613 \quad 0.612 \,/\, 0.615 \quad 0.614 \quad 0.617 \quad 0.616 \\ & \underline{0.651} \\ 100 & \underline{0.711} \\ & \underline{0.751} \qquad D^{ABCDEabcde}_{0\,1\,1\,\underline{0\,0}\,01\underline{001}} = BC\bar{a}b\,; \\ 1000 & \underline{1.611} \\ & \ldots . \end{array}$$

— *implicantul 8*

$$\begin{array}{rl} & \overset{1}{} \quad \overset{10}{} \quad \overset{20}{} \\ 40 & \underline{1.046} \,/\, 0.1.047 \,/\, 1.056 \quad 1.057 \,/\, 1.066 \quad 1.067 \quad 1.076 \quad 1.077 \\ & \underline{1.006} \\ 100 & \underline{1.146} \\ & \underline{1.106} \qquad D^{ABCDEabcde}_{1\,0\,\underline{001}\,0\,\underline{011}\,0} = A\bar{B}cd\,; \\ 200 & \underline{1.246} \\ & \ldots . \end{array}$$

— *implicantul 9*

$$\begin{array}{rl} & \overset{1}{} \quad \overset{2}{} \quad \overset{4}{} \\ 40 & \underline{1.051} \,/\, 1.050 \,/\, 1.053 \quad 1.052 \,/\, 1.055 \quad 1.054 \quad 1.057 \quad 1.056 \\ 100 & \underline{1.011} \\ & \underline{1.151} \\ 200 & \underline{1.111} \qquad D^{ABCDEabcde}_{1\,0\,\underline{001}\,01\underline{0\,0\,1}} = A\bar{B}\bar{a}b\,; \\ & \underline{1.251} \\ & \ldots . \end{array}$$

— *implicantul 10*

$$\begin{array}{rl} & \overset{1}{} \quad \overset{2}{} \quad \overset{4}{} \\ 40 & \underline{1.061} \,/\, 1.060 \,/\, 1.063 \quad 1.062 \,/\, 1.065 \quad 1.064 \quad 1.067 \quad 1.066 \\ 100 & \underline{1.021} \\ & \underline{1.161} \\ 200 & \underline{1.121} \qquad D^{ABCDEabcde}_{1\,0\,\underline{001}\,10\underline{0\,0\,1}} = A\bar{B}a\bar{b}\,; \\ & \underline{1.261} \\ & \ldots . \end{array}$$

— *implicantul 11*

$$\begin{array}{rl} & \begin{array}{cccccccc} 1 & & 2 & & & 10 & & \end{array} \\ 20 & \underline{1.105} \;/\; 1.104 \;/\; 1.107 \;\; 1.106 \;/\; 1.115 \;\; 1.114 \;\; 1.117 \;\; 1.117 \\ 40 & \underline{1.125} \\ & 1.145 \\ 200 & \underline{1.165} \\ & 1.305 \\ & 1.325 \\ & 1.345 \\ 400 & \underline{1.365} \\ & 1.505 \\ & \ldots \end{array}$$

$$D^{ABCDEabcde}_{1\,\underline{0\,0}\,1\,\underline{0\,0\,0}\,1\,\underline{0\,1}} = ADc;$$

— *implicantul 12*

$$\begin{array}{rl} & \begin{array}{cccccccc} 1 & & 2 & & & 4 & & \end{array} \\ 10 & \underline{1.203} \;/\; 1.202 \;/\; 1.201 \;\; 1.200 \;/\; 1.207 \;\; 1.206 \;\; 1.205 \;\; 1.204 \\ 40 & \underline{1.213} \\ & 1.243 \\ 100 & \underline{1.253} \\ & 1.303 \\ & 1.313 \\ & 1.343 \\ 400 & \underline{1.353} \\ & 1.603 \\ & \ldots \end{array}$$

$$D^{ABCDEabcde}_{1\,\underline{0}\,1\,\underline{0\,0}\,\underline{0\,0\,1\,1}} = AC\bar{a}.$$

Rezultă deci expresia logică minimală

$$F(Z) = ac\bar{A} \cup Cad \cup CDa\bar{b} \cup \bar{A}Bbc \cup \bar{A}Ba\bar{b} \cup BDbd \cup$$
$$\cup BC\bar{a}b \cup A\bar{B}cd \cup A\bar{B}\bar{a}b \cup A\bar{B}a\bar{b} \cup ADc \cup AC\bar{a}.$$

Problema tratată mai sus este luată din [72, p. 86—87]. Aici se arată că este posibilă doar o rezolvare aproximativă a acestei probleme, deducîndu-se o soluţie conţinînd 17 implicanţi, cu un total de 66 de variabile conţinute de aceştia.

Putem face deci constatarea că metoda simbolică de minimizare a permis rezolvarea precisă a acestei probleme, soluţia obţinută fiind verificată pentru toate stările definite de schema de comutaţie corespunzătoare simbolului de marcare dat. Această soluţie conţine

numai 12 implicanţi cu un total de numai 44 variabile conţinute de aceştia.

De altfel autorul lucrării citate remarcă că pentru un număr mai mare de nouă variabile binare este greu să avem siguranţa obţinerii unei forme minimale folosind metodele de minimizare cunoscute. Prin metoda simbolică de minimizare se extind deci posibilităţile de minimizare la un număr mai mare de variabile binare.

Important poate fi şi faptul că metoda simbolică de minimizare ar putea oferi un mijloc de determinare a algoritmului pentru astfel de probleme la programarea rezolvării acestora pe un calculator electronic numeric. Evident, folosirea calculatoarelor electronice numerice este indicată pentru probleme complexe cu un număr foarte mare de variabile de intrare.

Exemplul 6.8. Se cere minimizarea expresiei booleene definite prin simbolul de marcare

$$F(A, B, C) = D^{x_3x_2x_1x_0}_{(1,6,7,11,13,16,17),\ (4,5,6,7,12,14,15,16,17),\ (1,4,5,11,12,14,13,15,16,17).}$$

Folosind metoda simbolică de minimizare, se obţin succesiv implicanţii primi ai celor trei funcţii logice definite prin simbolul de marcare dat.

Pentru $F(A)$ se obţin astfel implicanţii

$$\begin{matrix}10\\1\,/\,11\end{matrix} \qquad D^{x_3x_2x_1x_0}_{\underline{0}\ 0\ 0\ 1} = \bar{x}_2\bar{x}_1x_0,$$

$$\begin{matrix}1\ \ 10\\6\,/\,7\,/\,16\ 17\end{matrix} \qquad D^{x_3x_2x_1x_0}_{\underline{0}\ 1\ 1\ \underline{0}} = x_2x_1,$$

$$\begin{matrix}4\\13\,/\,17\end{matrix} \qquad D^{x_3x_2x_1x_0}_{1\ \underline{0}\ 1\ 1} = x_3x_1x_0,$$

pentru F(B), obţinem

$$\begin{matrix}1\ \ 2\ \ 10\\4\,/\,5\,/\,6\ \ 7\,/\,14\ 15\ 16\ 17\end{matrix} \qquad D^{x_3x_2x_1x_0}_{\underline{0}\ 1\ \underline{0}\ \underline{0}} = x_2,$$

$$\begin{matrix}4\\12\,/\,16\end{matrix} \qquad D^{x_3x_2x_1x_0}_{1\ \underline{0}\ 1\ 0} = x_3x_1\bar{x}_0,$$

iar pentru $F(C)$ avem

$$\begin{matrix} 4 & 10 \\ 1 & 5 & 11 & 15 \end{matrix} \qquad D_{\underline{0}\ \underline{0}\ 0\ 1}^{x_3x_2x_1x_0} = \bar{x}_1x_0,$$

$$\begin{matrix} 10 & 1 \\ 4 & 14 & 5 & 15 \end{matrix} \qquad D_{\underline{0}\ 1\ 0\ \underline{0}}^{x_3x_2x_1x_0} = x_2\bar{x}_1,$$

$$\begin{matrix} 1 & 4 \\ 12 & 13 & 16 & 17 \end{matrix} \qquad D_{1\ \underline{0}\ 1\ \underline{0}}^{x_3x_2x_1x_0} = x_3x_1.$$

Formulele minimale rezultate sînt deci

$$F(A) = \bar{x}_2\bar{x}_1x_0 \cup x_2x_1 \cup x_3x_1x_0,$$

$$F(B) = x_2 \cup x_3x_1\bar{x}_0,$$

$$F(C) = \bar{x}_1x_0 \cup x_2\bar{x}_1 \cup x_3x_1.$$

Pentru comparație poate servi același exemplu, rezolvat prin metoda matriceală în [72, p. 112—115]. Problema folosirii în comun a unor termeni ai expresiilor logice se va pune mai tîrziu, cu ocazia sintezei unor astfel de circuite.

Scanned with OKEN Scanner

Capitolul 7 METODA SIMBOLICĂ DE ANALIZĂ A SCHEMELOR DE COMUTAȚIE

Problema analizei unei scheme de comutaţie date constă în determinarea funcţiilor pe care le poate îndeplini aceasta şi a depistării deficienţelor ce pot interveni în funcţionarea ei. Astfel, în cazul schemelor de comutaţie combinaţionale, fiind dată schema, se pune problema determinării diferitelor ei stări posibile, adică a determinării valorilor binare a variabilelor de ieşire care corespund tuturor combinaţiilor posibile ale valorilor binare ale variabilelor de intrare. De asemenea se poate cere depistarea posibilităţilor de hazard care intervin în funcţionarea schemei. În cazul schemelor de comutaţie secvenţiale se mai cere în plus descrierea secvenţelor posibile de lucru ale schemei.

Înainte de a trece la descrierea metodei simbolice de analiză a unor astfel de scheme să examinăm structura generală a unei scheme de comutaţie.

Într-o schemă de comutaţie se pot distinge patru părţi principale, şi anume : partea de comandă, partea de prelucrare a informaţiilor binare, partea de execuţie şi partea de semnalizare.

Partea de comandă conţine dispozitivele iniţiatoare de comenzi, cum ar fi diferitele butoane de comandă manuală, microîntrerupătoare, limitatoare de cursă sau contacte acţionate din exteriorul schemei considerate prin relee sensibile la anumiţi parametrii fizici. Variabilele binare care definesc această parte a schemei sînt variabile primare.

Partea de prelucrare a informaţiilor cuprinde relee şi contactoare sau module de comutaţie statică, prin care se prelucrează informaţiile primite de la partea de comandă şi cele primite de la partea de execuţie şi se elaborează în final semnale de comandă în stare să influenţeze partea de execuţie şi de semnalizare a schemei. Această parte îndeplineşte deci o anume funcţie logică.

Variabilele binare specifice acestei părţi a schemei de comutaţie sînt variabile primare şi variabilele de ieşire care au ataşate şi variabile secundare.

Scanned with OKEN Scanner

Partea de execuţie a schemelor de comutaţie cuprinde diferite elemente de execuţie, dispozitivele de cuplare la reţea a motoarelor electrice, dispozitive de pornire, oprire, reglaj de viteză sau contacte în stare să influenţeze starea anumitor elemente de execuţie care fac parte din sistemul discret de comandă automată. Specifice numai acestei părţi a schemei de comutaţie sînt numai variabilele de ieşire care nu au ataşate şi variabile secundare.

În fine, partea de semnalizare cuprinde diferite dispozitive optice sau acustice care servesc pentru semnalizarea îndeplinirii comenzilor date şi a stărilor schemei de comutaţie.

Primul pas în procesul de analiză a unei scheme de comutaţie constă în stabilirea funcţiilor logice aferente variabilelor de ieşire ale schemei. Exemplele de mai jos vor arăta cum se face acest lucru pentru diverse cazuri particulare. Această operaţie este comună tuturor metodelor de analiză folosite curent.

În cazul metodei simbolice de analiză, pasul următor în procesul de analiză constă în stabilirea simbolului de marcare pentru fiecare variabilă de ieşire separat. În acest scop se precizează mai întîi care dintre variabilele de intrare sînt primare şi care dintre acestea sînt secundare. Avînd făcută această precizare, se scrie şirul variabilelor primare succedat de cel al variabilelor secundare, întregul şir reprezentînd indicele superior al simbolului de marcare.

Pentru găsirea numerelor de stare aferente fiecărui simbol de marcare se impune scrierea formulei logice aferente lui în forma canonică disjunctivă. În acest scop se pot folosi regulile algebrei booleene, aşa cum s-a făcut la exemplul 6.6.

Mult mai expeditiv este însă procedeul următor, bazat pe formarea grupelor de numere de stare adiacente după metodica expusă în capitolul 6. Astfel se scriu mai întîi sub formă explicită toţi termenii expresiei logice al cărei simbol de marcare îl căutăm. Primul număr de stare al unui termen al expresiei logice se obţine înlocuind în şirul ordonat al variabilelor de intrare din indicele superior al simbolului de marcare variabilele directe conţinute de termen prin valoarea 1, celelalte variabile din şir fiind înlocuite cu valoarea „0". Numărul de stare rezultă prin convertirea în codul octal a numărului binar obţinut astfel.

Restul celorlalte numere de stare care definesc termenul respectiv vor corespunde tuturor adiacenţelor posibile. Pentru a le găsi se taie (prin subliniere) în şirul ordonat toate variabilele

conținute de termen. Valoarea octală a bitului 1, care corespunde cu locul fiecărei variabile rămase nesubliniate (netăiate) din acest șir, corespunde diferențelor de adiacență posibile pentru acest termen.

Se găsește astfel pentru numerele diferență de adiacență posibile mai întîi numărul de stare adiacent primului număr de stare găsit, apoi următoarele două numere de stare adiacente primelor, apoi următoarele patru numere de stare adiacente numerelor de stare găsite etc. Se procedează astfel, considerînd diferențele de adiacență posibile în ordinea lor crescătoare, pînă se epuizează toate diferențele de adiacență care afectează termenul respectiv al expresiei logice.

Simbolul de marcare aferent unei expresii logice, după cum rezultă din formula (3.8), va cuprinde cîte un reprezentant al tuturor numerelor de stare aferente termenilor conținuți de această formulă.

În cazul analizei schemei de contribuție statică, se poate folosi și o altă metodă de analiză, care va fi explicată prin exemplul 7.1.

Astfel, se precizează mai întîi ordinea variabilelor de intrare ale schemei, formînd astfel indicele superior al simbolului de marcare. Se găsesc apoi prin procedeul indicat mai sus numerele de stare aferente fiecărei variabile din acest șir, ca și cum ea ar reprezenta un termen. Folosind apoi formulele (3.7), (3.18) și (3.24) se găsesc succesiv numerele de stare asociate tuturor variabilelor de ieșire ale modulelor logice (elementelor de comutație statică) folosite în schemă. Pe această cale, din aproape în aproape se ajunge la ieșirile schemei, rezultînd în final numerele de stare asociate fiecărei variabile de ieșire a schemei de comutație. Cunoscînd aceste numere de stare putem scrie simbolul de marcare al întregii scheme.

Pentru schemele combinaționale de comutație, procesul de analiză se termină o dată cu găsirea simbolului de marcare aferent schemei. Într-adevăr, acest simbol ne indică combinațiile variabilelor de intrare pentru care variabilele de ieșire iau valoarea binară 1. Numerele de stare necuprinse de simbol ne indică combinațiile care corespund valorii 0 a variabilei de ieșire.

În cazul schemelor de comutație secvențiale, se cere în plus găsirea secvențelor de lucru ale schemei și ale stărilor ei de blocare, secvențe care pot evidenția și prezența hazardului în funcționarea schemei.

Secvențele de lucru se pot prezenta fie sub formă analitică, ca succesiune a stărilor schemei, așa cum s-a arătat în capitolul 2, fie sub formă grafică, adică sub forma diagramelor de faze.

Scanned with OKEN Scanner

Forma analitică de prezentare a secvențelor de lucru ale unei scheme de comutație secvențiale se preferă în cazul unui număr mic al variabilelor de intrare, iar cea grafică convine mai mult în cazul unui număr mai mare a acestor variabile. Forma grafică redă mai bine sub aspect fizic funcționarea schemei secvențiale. Este greu însă de a reda grafic toate secvențele posibile și stările care pot apărea în funcționarea schemei secvențiale. De aceea sub formă grafică se vor reda, de regulă, numai secvențele principale de lucru ale schemei. Secvențele secundare sau stările de blocare accidentale se pot deduce imediat, atunci cînd e necesar, folosind simbolul de marcare.

Exemplul 7.1. Să se facă analiza schemei sumatorului elementar al unui calculator numeric, schemă redată în figura 7.1.

Procedînd așa cum s-a arătat mai sus, se găsesc mai întîi numerele de stare aferente variabilelor x, y și z. Astfel, considerînd

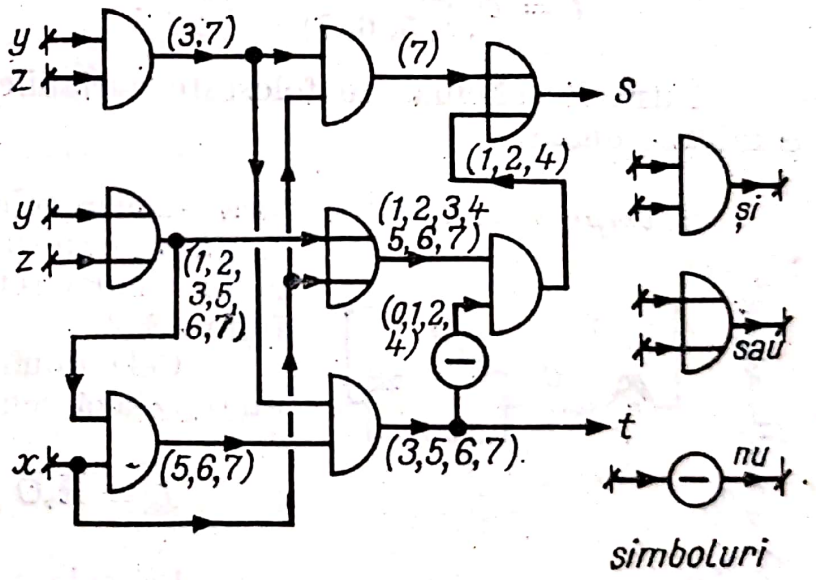


Fig. 7.1. — Analiza sumatorului elementar.

ordinea variabilelor xyz, primele numere de stare pentru variabilele x, y și z vor fi respectiv

$$(100)_8 = 4, \quad (010)_8 = 2 \quad \text{și} \quad (001)_8 = 1.$$

Diferențele de adiacență corespunzătoare celor trei variabile se deduc din

$$\underline{x}yz, \quad x\underline{y}z \quad \text{și} \quad xy\underline{z}.$$

Se obţin astfel numerele de stare

$$\begin{array}{l} \text{— pentru } x: \overset{1\ \ 2}{4\ /\ 5\ /\ 6\ \ 7}; \\ \text{— pentru } y: \overset{1\ \ 4}{2\ /\ 3\ /\ 6\ \ 7}; \\ \text{— pentru } z: \overset{2\ \ 4}{1\ /\ 3\ /\ 5\ \ 7}. \end{array}$$

Aplicînd apoi formulele (3.7), (3.18) şi (3.24), se găsesc numerele de stare la ieşirea tuturor modulelor logice folosite în schemă. Ele sînt date în figura 7.1.

Final se obţine

$$S = D^{xyz}_{(7)} \cup D^{xyz}_{(1,2,4)} = D^{xyz}_{(1,2,4,7)} = D^{xyz}_{151},$$

$$t = D^{xyz}_{(3,5,6,7)} = D^{xyz}_{27}.$$

Întrucît schema nu foloseşte variabile secundare, ea este pur combinaţională.

Exemplul 7.2. Să se facă analiza schemei de comandă automată a pornirii unui motor asincron cu rotor scurtcircuitat, schemă dată în figura 7.2.

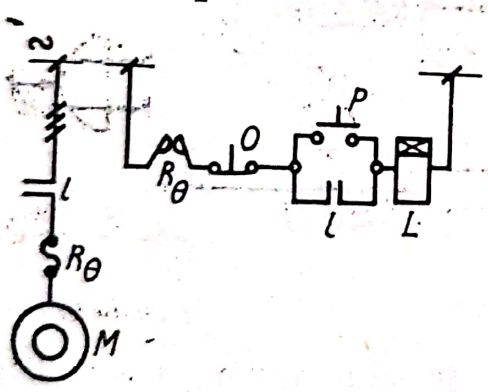


Fig. 7.2. — Schema analizată în exemplul 7.1.

Cele două funcţii logice care caracterizează schema dată, sînt

$$L = \overline{R}_\theta \overline{O}\,(P \cup l) \quad \text{şi} \quad M = l.$$

Primele numere de stare aferente termenilor $\overline{R}_\theta \overline{O} P$ şi $\overline{R}_\theta \overline{O} l$, în cazul cînd se alege următoarea ordine a variabilelor de intrare $R_\theta O P l$, se deduc astfel :

$$(0.010)_8 = 2 \quad \text{şi} \quad (0.001)_8 = 1.$$

Următoarele numere de stare ale celor doi termeni corespund diferenţelor de adiacenţă 1 şi 2, deduse din

$$\underline{R_\theta O P l} \quad \text{şi} \quad \underline{R_\theta O P \underline{l}}.$$

Astfel numerele de stare aferente celor doi termeni ai funcției logice L vor fi

— pentru $\overline{R}_0\overline{O}P \longrightarrow \begin{matrix}1\\2\end{matrix} / 3$;

— pentru $\overline{R}_0\overline{O}l \longrightarrow 1 / 3$.

Rezultă simbolurile de marcare

$$L = D^{R_0OPl}_{(1,2,3)} \quad \text{și} \quad M = D^{l}_{(1)}.$$

Întrucît schema folosește variabila l ca variabilă de reacție (secundară), ea este secvențială.

Secvențele principale de pornire și de oprire a motorului M se vor scrie, așa cum s-a arătat în capitolul 2,

$$\frac{0}{0} \xrightarrow{P=1} \frac{2}{1} \longrightarrow \frac{3}{1} \xrightarrow{P=0} \frac{1}{1} \quad \text{și:} \quad \frac{1}{1} \xrightarrow{O=1} \frac{5}{0} \longrightarrow \frac{4}{0} \xrightarrow{O=0} \frac{0}{0}.$$

În formă grafică, aceste secvențe sînt redate sub forma diagramei de faze din figura 7.3.

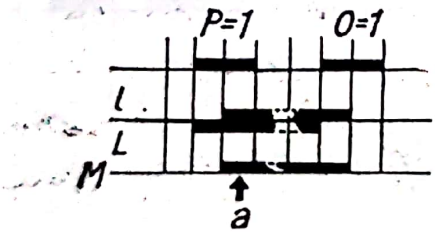


Fig. 7.3. — Diagrama de faze pentru exemplul 7.2.

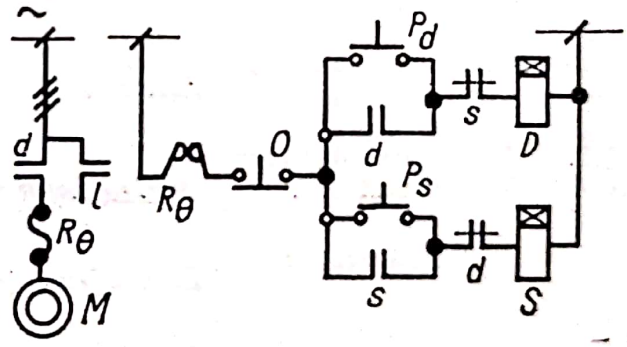


Fig. 7.4. — Schema analizată în exemplul 7.3.

Ne propunem să determinăm efectul intervenției releului termic ($R_0 = 1$) în starea (a) (fig. 7.3).

Starea (a) corespunde numărului de stare (3), pentru variabila L. Dacă $R_0 = 1$, ea trece în starea (13), care, după cum rezultă din simbolul de marcare, corespunde la $L = 0$. Întrucît imediat $l = 0$, această stare instabilă trece în starea stabilă (12), care corespunde la $L = M = 0$.

Exemplul 7.3. Să se facă analiza schemei de comandă automată reversibilă a unui motor asincron cu rotor scurtcircuitat, schemă redată în figura 7.4.

Se scriu mai întîi funcţiile logice care corespund celor trei variabile de ieşire:

$$D = \overline{R}_0 \overline{O} \overline{s} \, (P_d \cup d),$$

$$S = \overline{R}_0 \overline{O} \overline{d} \, (P_s \cup s),$$

$$M = d \cup s.$$

Aplicînd acelaşi procedeu ca în exemplul 7.2, găsim simbolurile de marcare

$$D = D^{R_0 O P_d sd}_{(1,4,5)}, \quad S = D^{R_0 O P_s sd}_{(2,4,6)} \text{ şi } \quad M = D^{sd}_{(1,2,3)}.$$

În figura 7.5. redăm sub forma diagramei de faze cele trei secvenţe principale (pornire în cele două sensuri şi oprire) de lucru ale schemei de comutaţie. Vom explica construcţia lor.

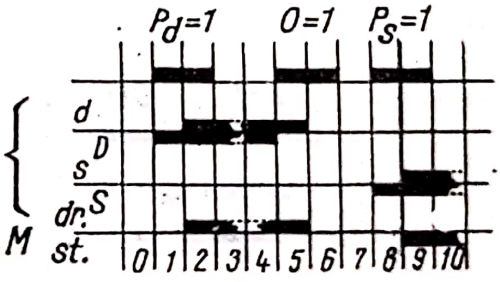


Fig. 7.5. — Diagrama de faze aferentă exemplului 7.3.

Iniţial vom trasa o singură linie orizontală, pe care se figurează comenzile date prin butoanele de comandă sau prin intervenţia releului termic. Partea de prelucrare a comenzilor va cuprinde două linii orizontale, pe care se vor figura stările binare ale variabilelor secundare şi a celor de ieşire de care sînt ataşate. În fine, partea de execuţie va fi reprezentată printr-o singură linie orizontală, pe care se va figura starea binară a motorului M şi sensul lui de rotaţie.

Secvenţa de pornire spre dreapta se iniţiază prin apăsarea butonului cu revenire P_d. Starea acestui buton influenţează numai

variabila D, al cărei simbol de marcare cuprinde starea (4), care corespunde la $D = 1$. Pentru poziția 1 din diagrama de faze vom reda numai valoarea binară $D = 1$.

Urmează apoi imediat $d = 1$. Variabila d afectează toate cele trei funcții logice, astfel încît în diagrama de faze vom reda valorile binare $D = 1$, $S = 0$ și $M = 1$ în poziția 2. Revenirea butonului P_d în stare normală nu modifică această stare stabilă.

Similar se determină secvența inițiată prin $P_s = 1$ și prin $O = 1$.

Dispunînd de patru variabile primare, sînt posibile 2^4 combinații ale lor, prin care se pot iniția diverse comenzi. Este dificil să le figurăm pe toate în diagrama de faze.

Secvențele posibile neredate în diagrama de faze rezultă însă imediat din simbolurile de marcare. Drept exemplu să considerăm că în starea redată în diagrama de faze prin poziția 3 apăsăm butonul P_s. Poziția 3 este caracterizată prin $d = D = M = 1$, restul variabilelor avînd valoarea binară 0. Variabila P_s este conținută numai de simbolul S ceea ce se poate constata examinînd indicii superiori ai simbolurilor de marcare. Pentru $P_s = d = R = 1$ și $O = s = 0$, avem $n = 15$ și rezultă $S = 0$, deci starea schemei în ansamblu nu se va modifica prin această comandă nouă. La fel se pot determina și alte stări de blocare sau secvențe secundare ale schemei analizate.

În cazul exemplului 7.3., diagrama de faze și diferitele stări posibile ale schemei se pot deduce și simplu din examinarea schemei reale redată în figura 7.4. Nu același lucru este posibil în cazul unor scheme care folosesc module de comutație statică sau în cazul unor scheme cu contacte complexe. În aceste cazuri, analiza schemei este mult ușurată prin folosirea metodei simbolice.

Exemplul 7.4. Să se facă analiza schemei de comandă automată a unui ascensor, schemă redată în figura 7.6.

Întrucît drept variabile secundare (de reacție) în schemă sînt folosite în afara variabilelor r și c și variabilele x, y și z, se impune determinarea funcțiilor logice și a variabilelor X, Y și Z, de care depind aceste ultime variabile secundare.

Rezultă următoarele cinci funcții logice deduse din schema de comutație dată:

$$R = \overline{L}_s \overline{c}(x \cup y \cup z),$$

$$C = \overline{L}_i \overline{L}_4 \overline{r} (J_1 \cup J_2 \cup J_3 \cup J_p \cup c),$$

$$X = \overline{L}_1 (S_1 \cup x),$$

$$Y = \overline{L}_2 (S_2 \cup y),$$

$$Z = \overline{L}_3 (S_3 \cup z).$$

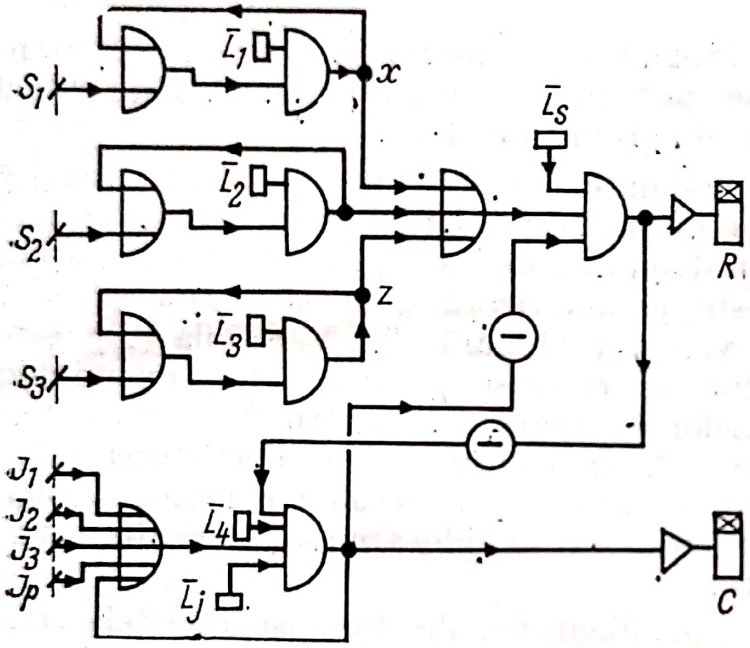


Fig. 7.6. — Comanda automată a unui ascensor (exemplul 7.4).

Simbolurile de marcare asociate lor vor fi

$$R = D^{L_i c x y z}_{(1,\ 2,\ \ldots 7)}; \quad X = D^{L_1 S_1 x}_{(1,\ 2,\ 3)},$$

$$C = D^{L_i L_4 r J_1 J_2 J_3 J_p c}_{(1,\ 2,\ \ldots 37)}; \quad Y = D^{L_2 S_2 y}_{(1,\ 2,\ 3)},$$

$$Z = D^{L_3 S_3 z}_{(1,\ 2,\ 3)}.$$

Ne propunem redarea în diagrama de faze numai a secvenței de urcare la etajul 2 și de coborîre la parter.

Comanda de urcare se dă prin apăsarea butonului cu revenire S_2. Pe parcurs sînt acționate traductoarele de poziție L_1 și L_2.

Abia acționarea lui L_2 determină dezexcitarea contactorului R, prin care este conectat la rețea motorul de acționare pentru sensul de rotație corespunzător urcării ascensorului.

Comanda de coborîre se dă prin apăsarea butonului cu revenire J_p. Pe parcurs sînt acționate traductoarele de poziție L_2 ($L_2 = 0$) și L_1($L_1 = 1$). Abia cînd ascensorul ajunge la parter, fiind acționat traductorul de poziție L_4, ascensorul se oprește. Controlul pozițiilor-limită sus și jos este făcut prin limitatoarele de cursă L_s și L_j.

Secvențele de lucru deduse folosind simbolurile de marcare sînt date în diagrama de faze din figura 7.7. Similar se pot determina

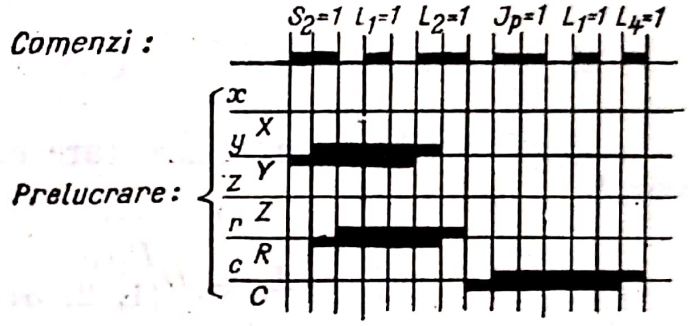


Fig. 7.7. — Diagrama de faze pentru exemplul 7.4.

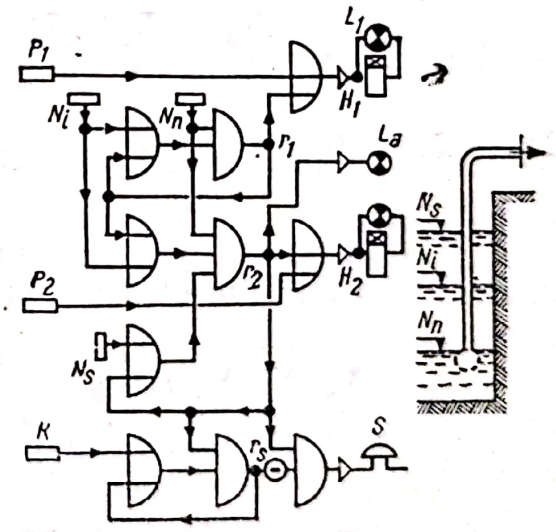


Fig. 7.8. — Comanda automată a unei instalații de pompare (exemplul 7.5).

și alte secvențe de lucru și stări de blocare care pot interveni în funcționarea ascensorului.

Exemplul 7.5. Să se facă analiza schemei de automatizare a unei instalații de pompare, schemă redată în figura 7.8.

Observarea schemei evidenţiază prezenţa a trei variabile secundare care nu sînt ataşate variabilelor de ieşire ale schemei, şi anume H_1, H_2 şi S. Pentru aceste variabile, notate în figura 7.8. prin r_1, r_2 şi r_s, vom lua în considerare variabilele de ieşire auxiliare R_1, R_2 şi R_s.

Din schema dată rezultă următoarele funcţii logice:

$$H_1 = L_1 = r_1 \cup P_1; \quad R_1 = N_n(N_i \cup r_1),$$

$$H_2 = L_2 = r_2 \cup P_2; \quad R_2 = L_a = N_n(r_1 \cup N_i)(N_s \cup r_2) =$$

$$S = r_2\bar{r}_s; \qquad = N_n r_1 N_s \cup N_n r_1 r_2 \cup$$

$$R_s = r_2(K \cup r_s); \qquad \cup N_n N_i N_s \cup N_n N_i r_2.$$

Simbolurile de marcare care corespund acestor funcţii logice vor fi

$$H_1 = L_1 = D^{P_1 r_1}_{(1,\ 2,\ 3)}; \quad S = D^{r_2 r_s}_{(2)},$$

$$H_2 = L_2 = D^{P_2 r_2}_{(1,\ 2,\ 3)}; \quad R_1 = D^{N_n N_i r_1}_{(5,\ 6,\ 7)},$$

$$R_2 = L_a = D^{N_n N_i N_s r_1 r_2}_{(23,\ 26,\ 27,\ 31,\ 33,\ 34,\ 35,\ 36,\ 37)}; \quad R_s = D^{K r_2 r_s}_{(3,\ 6,\ 7)}.$$

Redăm mai jos, schematizat, numai calculul făcut pentru determinarea numerelor de stare aferente funcţiei logice R_2:

$N_n\ N_s\ N_i\ r_1\ r_2$

$$- \cdot - - \cdot \rightarrow N_n r_1 N_i \rightarrow 26\ \overset{1}{/}\ 27\ \overset{10}{/}\ 36\ 37,$$

$$- \cdot \cdot - - \rightarrow N_n r_1 r_2 \rightarrow 23\ \overset{4}{/}\ 27\ \overset{10}{/}\ 33\ 37,$$

$$- - - \cdot \cdot \rightarrow N_n N_s N_i \rightarrow 34\ \overset{1}{/}\ 35\ \overset{2}{/}\ 36\ 37,$$

$$- - \cdot \cdot - \rightarrow N_n N_s r_2 \rightarrow 31\ \overset{2}{/}\ 33\ \overset{4}{/}\ 35\ 37.$$

Notațiile folosite sînt : N_n, N_i și N_s traductoare pentru nivelul normal, inferior și superior, iar P_1 și P_2 butoane ce oferă posibilitatea unor comenzi manuale.

Redăm mai jos schema calculului care a servit pentru determinarea secvenței de lucru a schemei. Numerotarea timpilor din diagrama de faze corespunde cu etapele calculului redat schematic mai jos :

Pentru a ușura exprimarea matematică, în indicele superior al simbolului D s-a trecut numărul binar ce corespunde combinației exprimate prin numărul de stare. Astfel, $R_1 = D^{100}$, semnifică valoarea binară a lui R_1 pentru $N_n = 1$ și $N_i = r_1 = 0$. Numărul de stare corespunzător este $(100)_8 = 4$ și corespunde, după cum se vede din simbolul de marcare asociat lui R_1, la $R_1 = 0$, întrucît acest simbol nu conține numărul de stare 4.

1) $N_n = 1$; $\rightarrow R_1 = D^{100} = 0$; $R_2 = L_a = D^{10 \cdot 000} = 0$,

2) $N_n = N_i = 1$; $\rightarrow R_1 = D^{110} = 1$; $R_2 = D^{11 \cdot 000} = 0$,

3) $N_n = N_i = 1$; $(r_1 = 1) \rightarrow H_1 = L_1 = D^{01} = 1$; $R_1 = D^{111} = 1$; $R_2 = L_a = D^{11 \cdot 010} = 0$,

4) $N_n = N_i = r_1 = 1$; $(N_s = 1) \rightarrow H_1 = L_1 = R_1 = 1$; $R_2 = L_a = D^{11 \cdot 110} = 1$,

5) $N_n = N_i = N_s = r_1 = 1$; $(r_2 = 1) \rightarrow H_1 = L_1 = R_1 = 1$; $H_2 = L_2 = D^{01} = 1$; $S = D^{10} = 1$; $R_2 = L_a = D^{11 \cdot 111} = 1$ $R_s = D^{010} = 0$,

6) $N_n = N_i = N_s = r_1 = r_2 = 1$; $(K = 1) \rightarrow H_1 = L_1 = R_1 = H_2 = L_2 = S = R_2 = L_a = 1$; $R_s = D^{110} = 1$,

7) $N_n = N_i = N_s = r_1 = r_2 = 1$; $(K = 0,\ r_s = 1) \rightarrow H_1 = L_1 = R_1 = H_2 = L_2 = R_2 = L_a = 1$; $S = D^{11} = 0$; $R_s = D^{011} = 1$,

8) $N_n = N_i = r_1 = r_2 = r_s = 1$; $(N_s = 0) \rightarrow H_1 = L_1 = R_1 = H_2 = L_2 = R_s = 1$; $R_2 = L_a = D^{11 \cdot 011} = 1$,

9) $N_n = r_1 = r_2 = r_s = 1$; $(N_i = 0) \rightarrow H_1 = L_1 = H_2 = L_2 = R_s = 1$; $R_1 = D^{101} = 1$; $R_2 = D^{10 \cdot 011} = 1$,

10) $r_1 = r_2 = r_s = 1$; $(N_n = 0) \rightarrow H_1 = L_1 = H_2 = R_s = 1$; $R_1 = D^{001} = 0$; $R_2 = D^{00 \cdot 011} = 0$,

11) $r_s = 1$; $(r_1 = r_2 = 0) \rightarrow H_1 = D^{00} = 0$; $H_2 = L_2 = D^{00} = 0$; $S = D^{01} = 0$; $R_1 = D^{000} = 0$; $R_2 = L_a = D^{00 \cdot 000} = 0$; $R_s = D^{001} = 0$.

Din diagrama de faze dedusă prin calculele de mai sus (fig. 7.9) rezultă următoarea secvență de lucru a instalației de pompe. Atît timp cît nivelul lichidului din bazinul sau din rezervorul deservit de pompe se găsește la o valoare normală ($N_n = 1$), cele două pompe acționate prin contactoarele H_1 și H_2 se găsesc în repaus.

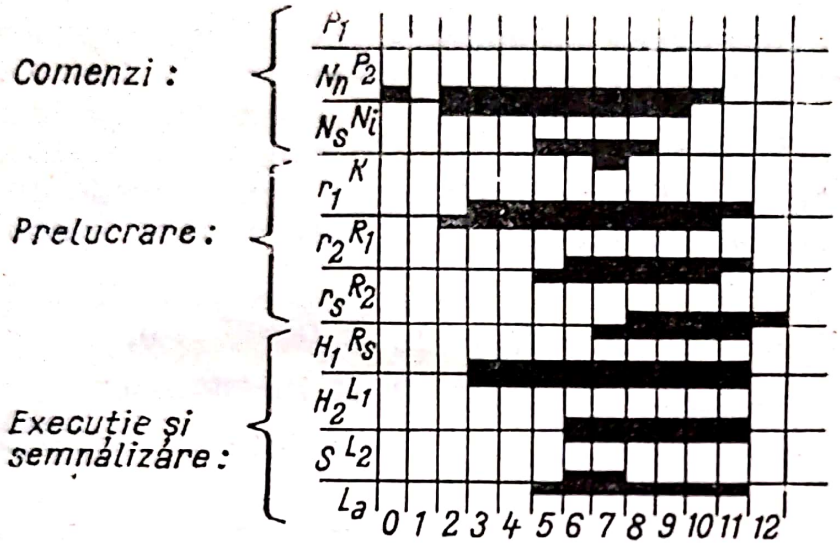


Fig. 7.9. — Diagrama de faze pentru exemplul 7.5.

Prin acumularea lichidului în bazin, nivelul crește și devine acționat și traductorul N_i (fig. 7.8). Prin aceasta se pune în funcțiune prima pompă ($H_1 = 1$) și se aprinde lampa semnalizatoare L_1 ($L_1 = 1$).

Dacă debitul de evacuare al primei pompe nu este suficient, atunci nivelul lichidului crește în continuare și la un moment dat este acționat traductorul de nivel N_s ($N_s = 1$). Prin aceasta se pune în funcțiune și a doua pompă prin intermediul contactorului H_2, se aprind lămpile semnalizatoare L_2 și L_a și sună soneria de alarmă S.

Prin intervenția operatorului uman ($K = 1$), soneria este oprită, continuînd însă să ardă lampa de semnalizare de avarie L_a. Lichidul este evacuat acum rapid prin debitul de evacuare sporit al celor două pompe, care continuă să funcționeze și după dezacționarea traductoarelor N_s și N_i. Abia traductorul N_n determină oprirea ambelor pompe și stingerea lămpilor semnalizatoare.

Schemele de comutație cu contacte conțin adesea și contacte cu temporizare mecanică. Temporizarea anclanșării acestor contacte este asigurată de diferite tipuri de mecanisme.

Scanned with OKEN Scanner

Variabilele asociate contactelor temporizate îndeplinesc rolul unor variabile primare, asigurînd tranzițiile dintre stările secvențelor de funcționare ale schemelor de comutație. Ele se vor specifica în simbolul de marcare printr-un indice atașat variabilelor temporizate.

Variabilele asociate contactelor cu temporizare se tratează ca variabile primare numai în cazul cînd schema de comutație conține și contacte netemporizate ale aceluiași contactor sau releu. În cazul releelor de timp, cînd toate contactele acestora sînt temporizate, variabilele respective, asociate contactelor, se tratează ca variabile secundare. În acest caz nu mai este necesară o discriminare între contactele temporizate și cele netemporizate, întrucît avem numai un singur tip de contacte pentru un releu sau contactor dat.

În general, analiza schemelor cu contacte oricît de complexe impune mai întîi scrierea funcțiilor logice pentru variabilele de ieșire ale schemei. Avînd cunoscute aceste formule, se determină ușor, prin procedeul exemplificat deja, numerele de stare aferente fiecărei variabile de ieșire.

Expresia logică corespunzătoare unei variabile de ieșire se scrie ca disjuncția tuturor funcțiilor logice ale ramurilor întîlnite plecînd de la contactor sau releu către celălalt pol de tensiune pe toate căile posibile. Vom arăta aceasta prin exemplele care urmează.

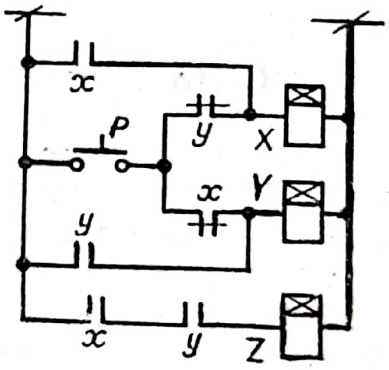


Fig. 7.10 — Schema analizată în exemplul 7.6.

Exemplul 7.6. Să se facă analiza schemei din figura 7.10, schemă care poate fi folosită pentru măsurarea timpilor de anclanșare (comutare) ai releelor și contactoarelor.

Funcțiile logice aferente celor trei variabile de ieșire se scriu

$$X = x \cup \overline{y}P \cup \overline{y}\overline{x}y = x \cup P\overline{y},$$

$$Y = y \cup \overline{x}P \cup \overline{x}\overline{y}x = y \cup P\overline{x},$$

$$Z = xy.$$

Pentru determinarea numerelor de stare aferente termenilor acestor formule, se întocmește următoarea schemă de calcul:

$$
\begin{array}{llll}
P\ x\ y & & & \overset{1}{2}\ /\ \overset{4}{3}\ /\ 6\ 7, \\
\cdot\ —\ \cdot & \to x & \to & \\
 & & & \overset{2}{1}\ /\ \overset{4}{3}\ /\ 5\ 7, \\
\cdot\ \cdot\ — & \to y & \to & \\
 & & & \overset{2}{4}\ /\ 6, \\
—\ \cdot\ — & \to P\bar{y} & \to & \\
 & & & \overset{1}{4}\ /\ 5, \\
—\ —\ \cdot & \to P\bar{x} & \to & \\
 & & & \overset{4}{3}\ /\ 7. \\
\cdot\ —\ — & \to xy & \to &
\end{array}
$$

Rezultă deci simbolul de marcare

$$F(Z, X, Y) = D^{Pxy}_{(2,\ 3,\ 4,\ 6,\ 7),\ (1,\ 3,\ 4,\ 5,\ 7),\ (3,\ 7)}.$$

Redarea secvențelor de lucru ale schemei poate fi făcută fie analitic, prin folosirea raportului de stare asociat schemei, fie grafic, prin trasarea diagramei de faze. În cazul unor scheme simple pentru care se poate stabili un raport de stare unic pentru întreaga schemă de comutație într-o formă ușor de exprimat analitic, se prefera prima cale mai simplă în acest caz.

În cazul nostru, secvențele inițiate prin apăsarea și eliberarea butonului P se vor stabili folosind raportul de stare

$$\frac{P(xy)}{Z(XY)} = \frac{n}{ZXY}.$$

Astfel, în cazul egalității timpilor de anclanșare ai contactoarelor X și $Y(\tau_x = \tau_y)$ se obține

$$\frac{0}{000} \xrightarrow{P=1} \frac{4}{110} \xrightarrow{\tau_x=\tau_y} \frac{7}{111} \xrightarrow{P=0} \frac{3}{111}.$$

În cazul inegalității timpilor de anclanșare $\tau_x \neq \tau_y$, se obține

$$(0) \xrightarrow{P=1} \frac{4}{110} \xrightarrow{\tau_x>\tau_y} \frac{5}{010} \xrightarrow{P=0} \frac{1}{010}$$

sau

$$(0) \xrightarrow{P=1} \frac{4}{110} \xrightarrow{\tau_x < \tau_y} \frac{6}{100} \xrightarrow{P=0} \frac{2}{100}.$$

Secvențele de mai sus evidențiază prezența hazardului de curse. Acest fenomen este nedorit în funcționarea schemelor de comutație. În cazul nostru însă el este folosit pentru reglarea timpului de comutare a unuia dintre contactoare sau relee. În acest scop se pot folosi trei lămpi semnalizatoare, branșate în derivație pe cele trei contactoare sau relee. Ele vor indica starea schemei în funcție de valoarea timpilor de comutare ai releelor sau contactoarelor X și Y. Unul dintre aceste aparate este luat drept etalon de timp de anclanșare.

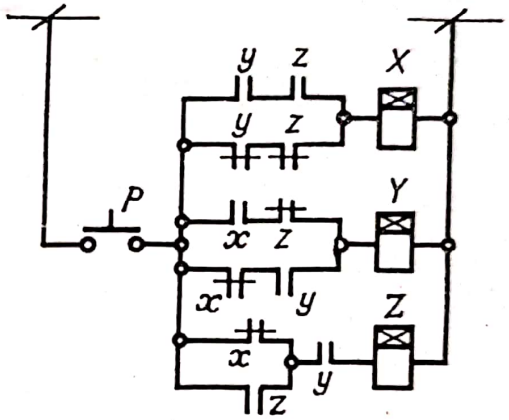


Fig. 7.11. — Schema analizată în exemplul 7.7.

Exemplul 7.7. Să se facă analiza schemei de comutație date în figura 7.11, schemă care poate fi folosită pentru acționarea ciclică a unui grup de trei electromotoare sau la reclame luminoase. Se presupune că timpii de anclanșare ai celor trei contactoare au fost reglați corespunzător cerințelor schemei.

Cele trei funcții logice ale schemei se vor scrie astfel:

$$X = Pyz \cup P\bar{y}\bar{z} = D^{Pxyz}_{(10,\ 13,\ 14,\ 17)},$$

$$Y = Px\bar{z} \cup P\bar{x}y = D^{Pxyz}_{(12,\ 13,\ 14,\ 16)},$$

$$Z = P\bar{x}y \cup Pyz = D^{Pxyz}_{(12,\ 13,\ 17)}.$$

Simbolul de marcare rezultant va fi

$$F(X, Y, Z) = D^{Pxyz}_{(10,\ 13,\ 14,\ 17)\ (12,\ 13,\ 14,\ 16),\ (12,\ 13,\ 17)}.$$

Raportul de stare al schemei fiind

$$\frac{P(xyz)}{XYZ} = \frac{n}{XYZ},$$

rezultă următoarea secvenţă de lucru a schemei iniţiată, prin închiderea întrerupătorului P:

$$\frac{0}{000} \xrightarrow{P=1} \frac{10}{100} \to \frac{14}{110} \to \frac{16}{010} \to \frac{12}{011} \to \frac{13}{111} \to \frac{17}{101} \to \frac{15}{000} \to (10) \ldots$$

Exemplul 7.8. Să se facă analiza schemei combinaţionale din figura 7.12.

Funcţia logică a schemei se scrie

$$X = c\bar{b}\bar{a} \cup \bar{c}ba \cup cb\bar{a} \cup c\bar{b}a = D^{abc}_{(0,\,3,\,5,\,6)}.$$

Exemplul 7.9. Să se facă analiza schemei combinaţionale din figura 7.13.

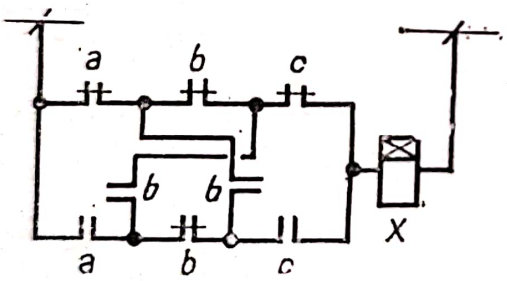


Fig. 7.12. — Schema analizată în exemplul 7.8.

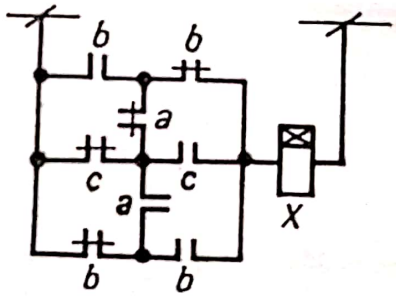


Fig. 7.13. — Schema analizată în exemplul 7.9.

Funcţia logică a schemei se scrie

$$X = \bar{b}\bar{a}c \cup c\bar{a}b \cup ca\bar{b} \cup ba\bar{c} = D^{abc}_{(0,\,3,\,5,\,6)}.$$

Remarcăm că întrucît simbolurile de marcare ale schemelor din figurile 7.12 şi 7.13 sînt identice, cele două scheme sînt funcţional echivalente. Exemplele 7.8 şi 7.9 au fost alese pentru a arăta cum prin marcarea schemelor cu ajutorul simbolului de marcare putem uşor descoperi identitatea funcţională a unor scheme diferite ca structură fizică şi aparent diferite şi ca funcţionare.

Scanned with OKEN Scanner

Capitolul 8

ANALIZA SCHEMEI DE AUTOMATIZARE A UNEI CARDE DE BUMBAC

În cele ce urmează vom expune metodica folosită pentru analiza unei maşini automate complexe, acest mod de analiză putînd servi drept ghid pentru alte cazuri similare.

Pentru a face posibilă analiza acestui sistem automat, a fost necesar să atașăm fiecărui buton sau cheie de comandă cîte o variabilă binară. Această variabilă are valoarea binară 1, dacă butonul sau cheia de comandă sînt acţionate şi are valoarea binară 0, dacă acestea nu sînt acţionate.

Semnificaţia notaţiilor făcute în figura 8.1, care redă panoul de comandă şi semnalizare optică, este : O_p-buton pentru comanda opririi perietorului cardei, P_i-buton pentru comanda mişcării perietorului cu viteză mică, P_r-buton pentru comanda mişcării perietorului cu viteză mare (de regim), O_m-buton de comandă a opririi motorului principal de acţionare, K_s-cheie de sens şi K_a-cheie de automatizare prin care maşina poate fi făcută să funcţioneze automat sau cu comandă manuală (neautomat).

Comanda de oprire a perietorului se poate face din două locuri şi anume : din faţă şi din spatele cardei, folosind în acest scop butoanele notate cu O_{ps} şi O_{pf}. Practic însă, starea binară a acestei comenzi este descrisă de variabila $O_p = O_{ps} = O_{pf}$.

Starea lămpilor de semnalizare optică este descrisă prin variabilele binare L_r-lampă roşie, care indică perietorul oprit, L_g-lampă galbenă, care indică starea de viteză redusă a perietorului, L_v-lampă verde, care indică sulul micşorat, L_{b1}-lampă albastră, care indică pătura de bumbac prea groasă, L_{b2}-lampă albastră, care indică ruperea benzii la cană, L_{b3}-lampă albastră, care indică atingerea lungimii programate a benzii din cană, L_m-lampă cu neon, care indică starea de funcţionare a motorului principal şi L-lampă cu neon, care indică prezenţa tensiunii electrice de alimentare.

Lampa L_{r1}-arată dacă perietorul este în repaos sau se roteşte cu viteză mică iar lampa L_{a1}-indică sulul cu pătură de bumbac

micşorat. Aceste două lămpi sînt montate în partea superioară a cardei şi pot fi observate de la distanţă de personalul de deservire.

Carda de bumbac Platts, care face obiectul acestei analize, este acţionată de un motor principal M_p, care antrenează perietorul şi tamburul principal al cardei şi de un servomotor M_s, care este necesar pentru schimbarea variatorului de viteză pentru perietor.

Schema monofilară a circuitului de forţă a cardei este dată în figura 8.2.

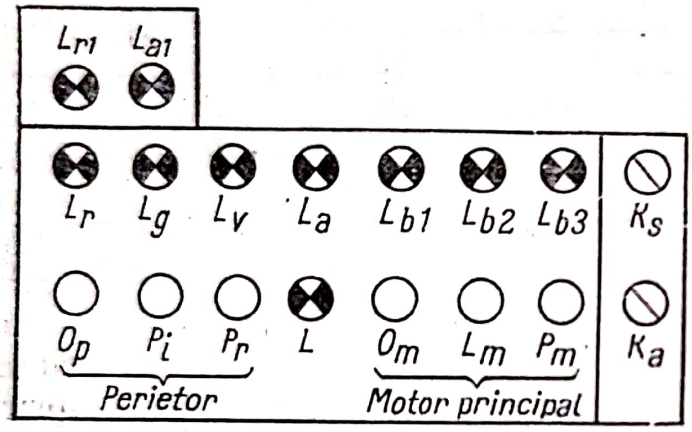


Fig. 8.1. — Panoul de comandă şi semnalizare optică.

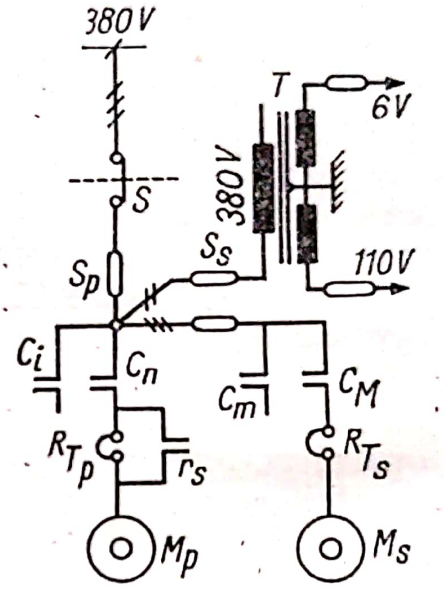


Fig. 8.2. — Schema de acţionare electrică a cardei.

În structura acestui circuit intră separatorul S, siguranţele fuzibile de protecţie la suprasarcini mari de scurtă durată S_p, releele termice R_{Tp} şi R_{Ts}, pentru protecţia la suprasarcini mici, dar de lungă durată, contactele contactoarelor C_n şi C_i, care conectează motorul principal al cardei la reţeaua de alimentare pentru cele două sensuri posibile de rotaţie, normal (C_n) şi invers (C_i), contactele r_s ale releului R_s de şuntare a rezistoarelor R_{Tp} pe durata pornirii motorului şi contactele contactoarelor care cuplează servomotorul M_s, pentru viteza mare a perietorului (C_M) şi pentru viteza mică a acestuia (Cm). Grupul de contacte c_m şi c_i, asigură o succesiune inversă a fazelor de alimentare a celor două motoare, în raport cu succesiunea fazelor asigurată prin contactele c_n şi c_M.

Transformatorul T, furnizează prin secundarele sale cele două tensiuni necesare alimentării schemei de comandă automată a cardei.

Cuplarea motorului M_p cu perietorul este asigurată prin intermediul cuplei electromagnetice K_e (fig. 8.3), alimentate cu tensiunea redresată de redresorul R_d. Reglarea cuplei electromagnetice (a curentului ei) se face prin reostatul R_h.

Controlul automat al vitezei perietorului este efectuat prin intermediul microîntrerupătoarelor M_m și M_M, care limitează deplasarea lanțului pe discurile conice ale variatorului de viteză cu ajutorul servomotorului M_s, pozițiile limită pentru viteza mică fiind controlate de M_m, iar pentru viteza mare, de M_M.

Controlul grosimii păturii de bumbac alimentate este efectuat prin microîntrerupătoarele M_g, controlul lungimii benzii de bumbac debitate în cană este efectuat prin microîntrerupătorul M_l, iar micșorarea sulului sub limita permisă este sesizată prin microîntrerupătorul M_s. Traductorul cu role T_r, este folosit pentru a sesiza ruperea sau terminarea benzii debitate, precum și abaterile inadmisibile ale grosimii benzii.

Pentru executarea comenzilor date de la panoul de comandă și pentru prelucrarea logică a tuturor informațiilor care intervin în funcționarea cardei, sînt folosite un număr de 15 relee și contactoare, cărora li se atașează variabilele binare date mai jos:

C_n și C_i-contactorul de sens normal și de sens invers pentru cuplarea la rețea a lui M_p.

C_m și C_M-contactorul de viteză mică și de viteză mare, pentru cuplarea la rețea a lui M_s.

R_s — releu de șuntare a rezistoarelor releului termic R_{Tp}.
R_t — releu de timp.
R_c — releul contorului de lungime a benzii debitate în cană.
R_p — releul de cuplare a perietorului.
R_m — releu pentru viteza mică a perietorului.
R_g — releu pentru depășirea grosimii permise a păturii alimentate.
R_{bm} — releul butonului de viteză mică a perietorului.
R_a — releul de alimentare a circuitului de semnalizare optică.
R_M — releul pentru viteza mare a perietorului.
R_{sm} — releul sulului cu pătură micșorat.
R_{rc} — releu pentru rolele traductorului T_r.

Scanned with OKEN Scanner

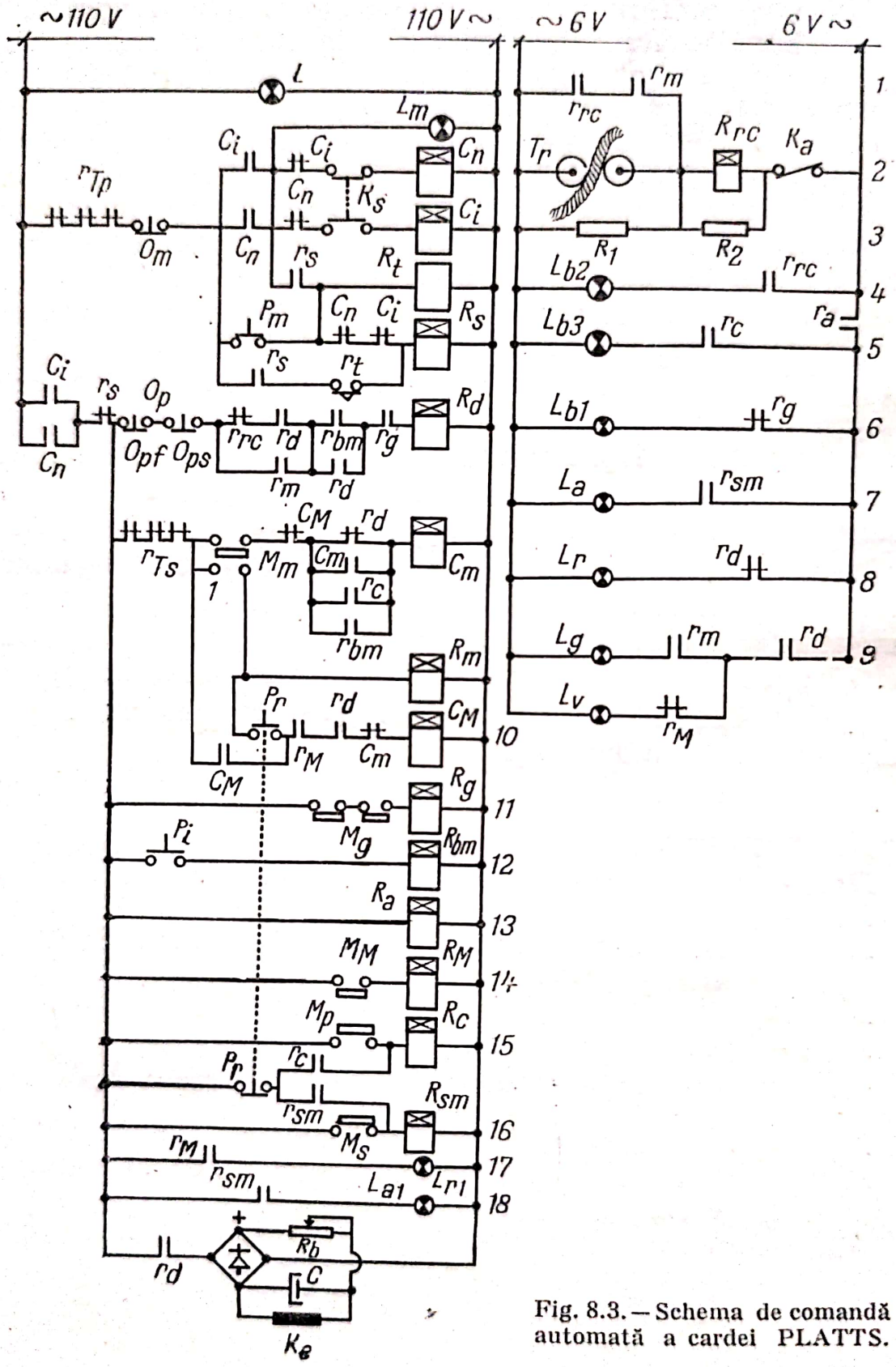


Fig. 8.3. — Schema de comandă automată a cardei PLATTS.

Scanned with OKEN Scanner

Toate contactele acestor contactoare și relee electromagnetice se vor nota prin literele mici corespunzătoare, păstrînd neschimbați indicii.

Cele două părți ale schemei de comandă automată a cardei Platts sînt date în figura 8.3.

Partea de comandă automată a cardei cuprinde contactele releelor termice și cele acționate manual sau pe cale mecanică; partea de prelucrare logică a informațiilor este reprezentată de cele 15 relee și contactoare electromagnetice; partea de semnalizare optică este reprezentată de lămpile semnalizatoare, iar partea de execuție este reprezentată de cupla electromagnetică K_e, servomotorul M_s și motorul principal M_p. Cele două sensuri posibile de rotație ale electromotoarelor le vom reprezenta prin variabilele binare M_{pn} și M_{pi}, respectiv: M_{sm} și M_{sM}.

Convenim ca starea diferitelor părți ale automatului finit reprezentat de schema analizată, în diferite momente de timp, să o redăm în formă codificată prin numere exprimate în codul octal. În acest scop, formăm șirurile de variabile care definesc starea celor patru părți ale automatului. Aceste șiruri sînt următoarele:

Partea de comandă (C)

$$O_p P_i P_r \cdot O_m P_m K_s \cdot K_a r_{Tp} r_{Ts} \cdot M_m M_M M_g \cdot M_1 M_s T_r .$$

Partea de prelucrare logică (PL)

$$\frac{c_n c_i r_t \cdot r_s r_p c_m \cdot r_m c_M r_g \cdot r_{tm} r_{aM} \cdot r_c r_{sm} r_{rc}}{C_n C_i R_t \cdot R_s R_p C_m \cdot R_m C_M R_g \cdot R_{tm} R_a R_M \cdot R_c R_{sm} R_{rc}} .$$

Partea de semnalizare optică (S)

$$LL_m \cdot L_{r1} L_{a1} L_v \cdot L_g L_r L_a \cdot L_{b1} L_{b2} L_{b3} .$$

Partea de execuție (E)

$$M_{pn} M_{pi} \cdot M_{sm} M_{sM} K_e .$$

Înlocuind cu 1 sau 0 variabilele binare din aceste șiruri se obțin numere binare, care apoi pot fi ușor convertite în numere octale. Raportul de stare care folosește aceste numere poate reda în formă codificată diferitele secvențe de lucru ale automatului.

În scopul găsirii acestor secvenţe ne putem folosi fie de schema de comutaţie (fig. 8.3) fie de expresiile logice care descriu funcţiile logice ale diferitelor elemente ale acestei scheme, expresii care se pot scrie imediat folosind această schemă. Noi vom alege calea a doua, pe care o considerăm mai uşoară în cazul schemelor de comutaţie dinamică. În cazul schemelor de comutaţie statică, calea a doua este cea mai indicată şi uneori cînd schema este complexă, unica cale posibilă de analiză completă.

În funcţionarea cardei Platts, distingem următoarele secvenţe principale de funcţionare :

a) SECVENŢA DE PORNIRE A MOTORULUI PRINCIPAL AL CARDEI

Comanda de pornire a motorului principal al cardei se dă prin apăsarea butonului de comandă P_m. Prin aceasta, după cum se vede din schema dată în figura 8.3, este pusă sub tensiune bobina releului de şuntare R_s şi cea a releului de timp R_t.

Codificat, aceasta se exprimă prin stările următoare :

$$\frac{0}{0} \xrightarrow{P_m=1} \frac{0}{14.000}$$

Contactele r_s se închid înaintea contactelor r_t şi prin aceasta este pusă sub tensiune bobina contactorului de sens normal C_n a motorului M_p. Schema trece prin stările

$$\frac{0}{14.000} \xrightarrow{r_s=1} \frac{4.000}{54.000}.$$

După închiderea contactelor c_n ale contactorului C_n, putem ridica mîna de pe butonul P_m, starea schemei nemodificîndu-se pînă cînd are loc anclanşarea contactelor temporizate r_t. Acestea determină, după cum se deduce din schemă, dezexcitarea bobinei releului R_s. Codificat, aceasta se scrie astfel

$$\frac{4.000}{54.000} \xrightarrow{c_n=1} \frac{44.000}{54.000} \xrightarrow{P_m=0} \frac{44.000}{54.000} \xrightarrow{r_t=1} \frac{54.000}{50.000}.$$

Dezanclanşarea contactelor r_s, după cum rezultă din analiza schemei din figura 8.3, întrerupe curentul bobinelor R_t şi R_s prin intermediul contactelor de pe liniile 4, şi 5, iar prin închiderea

contactului de pe linia 6, pune sub tensiune releele: C_m, R_g, R_a și R_M. Codificat, aceasta se exprimă astfel

$$\frac{54.000}{50.000} \xrightarrow{r_s=0} \frac{50.000}{41.130}.$$

Anclanșarea sau dezanclanșarea contactelor acestor relee și contactoare nu modifică starea schemei, adică

$$\frac{50.000}{41.130} \xrightarrow{r_t=0;\ c_m=r_g=r_a=r_M=1} \frac{41.130}{41.130}.$$

Ieșirea schemei din starea stabilă, exprimată prin ultimul raport, nu este posibilă decît prin intervenția unei variabile din partea de comandă a schemei. Această variabilă descrie starea microîntrerupătorului M_m, care controlează poziția lanțului pe discurile conice ale variatorului de viteză. În momentul cînd starea variatorului corespunde vitezei mici, devine acționat M_m (de pe liniile 7—8) care trece pe poziția 1, punînd sub tensiune bobina releului R_m și întrerupînd curentul prin C_m.

Codificat, aceasta se scrie

$$\frac{41.130}{41.130} \xrightarrow{M_m=1} \frac{41.130}{40.530} \xrightarrow{r_m=1;\ c_m=0} \frac{40.530}{40.530}.$$

Se ajunge astfel într-o nouă stare stabilă, pregătitoare altei secvențe de lucru a mașinii.

Evoluția în timp a părții de comandă, semnalizare și execuție a schemei de comandă automată, pe durata secvenței de mai sus, se descrie codificat astfel:

$$C: \quad 0 \xrightarrow{P_m=1} 2.000 \xrightarrow{P_m=0} 0 \xrightarrow{M_m=1} 40$$

$$S: \quad 2.000 \xrightarrow{r_s=1} 3.000 \xrightarrow{r_a=r_M=1} 3.420$$

$$E: \quad 0 \xrightarrow{c_n=1} 20 \xrightarrow{c_m=1} 24 \xrightarrow{c_m=0} 20$$

Urmărind toate schimbările variabilelor notate în secvența de mai sus (deasupra săgeților) și analizînd modificările determinate de acestea în partea de comandă, semnalizare și execuție, se deduc ușor stările descrise de tranzițiile de mai sus.

Scanned with OKEN Scanner

Presupunînd că deducerea stărilor secvenţei de pornire a motorului M_p rezultă din prezentarea analitică de mai sus, vom interpreta doar starea finală a cardei.

Astfel, numărul $40.530_{(8)}$ ne arată că în urma secvenţei iniţiată prin apăsarea butonului P_m, care e suficient a fi apăsat numai pînă anclanşează releul R_s, devin în final acţionate contactoarele şi relecle C_n, R_m, R_g, R_a şi R_M. Numărul $40_{(8)}$ redă în formă codificată faptul că prin acţionarea lui P_m şi apoi dezacţionarea sa, avem: $M_m = 1$.

Numărul $3.420_{(8)}$, ne arată că în final sînt aprinse lămpile semnalizatoare L, L_m, L_{r1} şi L_r, iar numărul $10_{(8)}$, ne arată că în final motorul M_p se roteşte în sensul normal, adică $M_{pn} = 1$.

Similar se interpretează şi celelalte secvenţe şi stări finale, date mai jos.

b) SECVENŢA DE PORNIRE A PERIETORULUI CU VITEZA MICĂ

$$PL: \frac{40.530}{40.530} \xrightarrow{P_i=1} \frac{40.530}{40.570} \xrightarrow{r_{bm}=1} \frac{40.570}{42.570} \xrightarrow{r_p=1} \frac{42.570}{42.570} \xrightarrow{P_i=0} \frac{42.570}{42.530} \xrightarrow{r_{bm}=0} \frac{42.530}{42.530}.$$

$$S: \quad 3.420 \xrightarrow{r_p=1} 3.440.$$

$$E: \quad 10 \xrightarrow{r_p=1} 11.$$

c) SECVENŢA DE PORNIRE A PERIETORULUI CU VITEZĂ MARE

$$PL: \frac{42.530}{42.530} \xrightarrow{P_r=1} \frac{42.530}{42.730} \xrightarrow{c_M=1} \frac{42.730}{42.730} \xrightarrow{P_r=0} \frac{42.730}{42.730} \xrightarrow{M_m=0} \frac{42.730}{42.330} \xrightarrow{r_m=0} \frac{42.330}{42.330} \xrightarrow{M_M=1} \frac{42.330}{42.320} \xrightarrow{r_M=0} \frac{42.320}{42.120} \xrightarrow{c_M=0} \frac{42.120}{42.120}.$$

$$S: \quad 3.440 \xrightarrow{r_m=0} 3.400 \xrightarrow{r_M=0} 3.100.$$

$$E: \quad 11 \xrightarrow{c_M=1} 15 \xrightarrow{c_M=0} 11.$$

Scanned with OKEN Scanner

d) SECVENȚA DE OPRIRE A PERIETORULUI

$$PL: \frac{42.120}{42.120} \xrightarrow{o_p=1} \frac{42.120}{42.120} \xrightarrow{r_p=0} \frac{40.120}{42.120} \xrightarrow{o_p=0} \frac{40.120}{41.120} \xrightarrow{c_m=1}$$

$$\frac{41.120}{41.120} \xrightarrow{M_m=0} \frac{41.120}{41.130} \xrightarrow{r_M=1} \frac{41.130}{41.130} \xrightarrow{M_m=1}$$

$$\frac{41.130}{40.530} \xrightarrow{c_m=0;\ r_m=1} \frac{40.530}{40.530}.$$

$$S: \quad 3.100 \xrightarrow{r_p=0} 3.020 \xrightarrow{r_M=1} 3.420.$$

$$E: \quad 11 \xrightarrow{r_p=0} 10 \xrightarrow{c_m=1} 12 \xrightarrow{c_m=0} 10.$$

e) ATINGEREA LUNGIMII PROGRAMATE A BENZII DEPUSE ÎN CANĂ

$$PL: \frac{42.120}{42.120} \xrightarrow{M_l=1} \frac{42.120}{42.124} \xrightarrow{r_c=1} \frac{42.124}{43.124} \xrightarrow{M_l=0} \frac{42.124}{43.124} \xrightarrow{c_m=1}$$

$$\frac{43.124}{43.124} \xrightarrow{M_M=0} \frac{43.124}{43.134} \xrightarrow{r_M=1} \frac{43.134}{43.134} \xrightarrow{M_m=1}$$

$$\frac{43.134}{42.534} \xrightarrow{c_m=0;\ r_m=1} \frac{42.534}{42.534}.$$

$$S: \quad 3.100 \xrightarrow{r_c=1} 3.101 \xrightarrow{r_M=1} 3.401 \xrightarrow{r_m=1} 3.441.$$

$$E: \quad 11 \xrightarrow{c_m=1} 15 \xrightarrow{r_m=0} 11.$$

Scanned with OKEN Scanner

Dacă se înlocuiește cana și se apasă pe butonul P_r, se obține secvența :

$$PL: \frac{42.534}{42.534} \xrightarrow{P_r=1} \frac{42.534}{42.730} \xrightarrow{r_c=0;\ c_M=1} \frac{42.730}{42.730} \xrightarrow{P_r=0}$$

$$\frac{42.730}{42.730} \xrightarrow{M_m=0} \frac{42.730}{42.330} \xrightarrow{r_m=0} \frac{42.330}{42.330} \xrightarrow{M_M=1}$$

$$\frac{42.330}{42.320} \xrightarrow{r_M=0} \frac{42.320}{42.120} \xrightarrow{c_M=0} \frac{42.120}{42.120}.$$

$$S: \quad 3.441 \xrightarrow{r_c=0} 3.440 \xrightarrow{r_m=0} 3.400 \xrightarrow{r_M=0} 3.100.$$

$$E: \quad 11 \xrightarrow{c_M=1} 15 \xrightarrow{c_M=0} 11.$$

f) DEPĂȘIREA GROSIMII PĂTURII

$$PL: \frac{42.120}{42.120} \xrightarrow{M_g=1} \frac{42.120}{42.020} \xrightarrow{r_g=0} \frac{42.020}{42.020} \xrightarrow{r_p=0} \frac{40.020}{41.020} \xrightarrow{c_m=1}$$

$$\frac{41.020}{41.020} \xrightarrow{M_M=0} \frac{41.020}{41.030} \xrightarrow{r_M=1} \frac{41.030}{41.030} \xrightarrow{M_m=1}$$

$$\frac{41.030}{40.430} \xrightarrow{c_m=0;\ r_m=1} \frac{40.430}{40.430}.$$

$$S: \quad 3.100 \xrightarrow{r_g=0} 3.104 \xrightarrow{r_p=0} 3.024 \xrightarrow{r_M=1} 3.424.$$

$$E: \quad 11 \xrightarrow{r_p=0} 10 \xrightarrow{c_m=1} 12 \xrightarrow{c_m=0} 10.$$

Scanned with OKEN Scanner

g) MICȘORAREA SULULUI CU PĂTURĂ SUB LIMITA PERMISĂ

$$PL: \frac{42.120}{42.120} \xrightarrow{M_s=1} \frac{42.120}{42.122} \xrightarrow{r_{sm}=1} \frac{42.122}{42.122} \xrightarrow{M_s=0} \frac{42.122}{42.122}.$$

$$S: \quad 3.100 \xrightarrow{r_{sm}=1} 3.310; \quad E: \ 11.$$

h) INTERVENȚIA TRADUCTORULUI CU ROLE DE LA CANĂ

$$PL: \frac{42.122}{42.122} \xrightarrow{T_r=1} \frac{42.122}{42.123} \xrightarrow{r_{rc}=1} \frac{42.123}{40.123} \xrightarrow{r_p=0}$$

$$\frac{40.123}{41.123} \xrightarrow{c_m=1} \frac{41.123}{41.123} \xrightarrow{M_M=0} \frac{41.123}{41.133} \xrightarrow{r_M=1} \frac{41.133}{41.133} \xrightarrow{M_m=1}$$

$$\frac{41.133}{40.533} \xrightarrow{c_m=0;\ r_m=1} \frac{40.533}{40.533} \xrightarrow{T_r=0} \frac{40.533}{40.533}.$$

$$S: \quad 3.310 \xrightarrow{r_{rc}=1} 3.312 \xrightarrow{r_p=0} 3.232 \xrightarrow{r_M=1} 3.632.$$

$$E: \qquad 11 \xrightarrow{r_p=0} 10 \xrightarrow{c_m=1} 12 \xrightarrow{c_m=0} 10.$$

i) OPRIREA MAȘINII

$$PL: \frac{42.120}{42.120} \xrightarrow{O_m=1} \frac{42.120}{2.120} \xrightarrow{c_n=0} \frac{2.120}{0} \longrightarrow 0.$$

$$S: \qquad 11 \xrightarrow{c_n=0} 10. \quad E: \ 0.$$

Scanned with OKEN Scanner

Capitolul 9

TRANSFORMAREA SCHEMELOR CU CONTACTE

Procesul de sinteză a unei scheme de comutație ne conduce, de regulă, la structurarea (realizarea practică) unei scheme π (fig. 9.1 *a*). În acest caz, starea aparatelor sau a dispozitivelor C_n, depinde de funcțiile de lucru f_n ale circuitelor de comutație care controlează starea acestora.

Adesea, prin transformarea schemei π într-o schemă echivalentă stelată (fig. 9.1*b*), în triunghiu simplu sau multiplu (fig. 9.1*c*) ori în punte (fig. 9.1*d*) se poate obține o realizare mai eco-

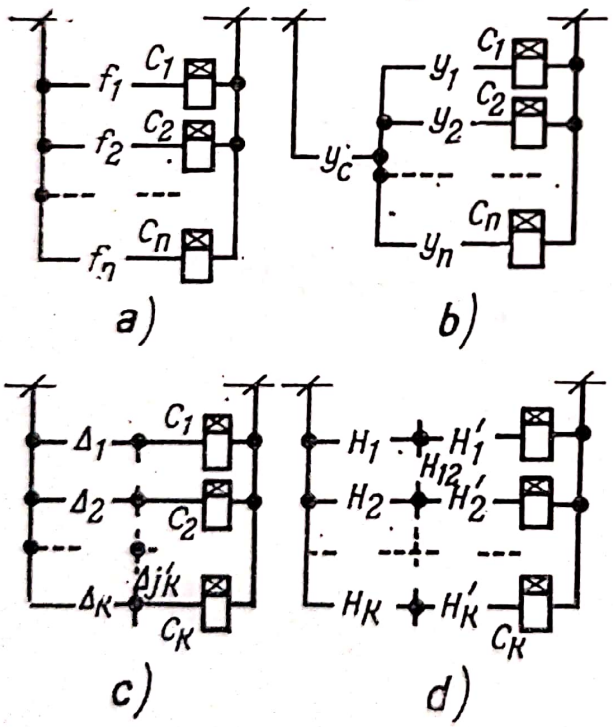


Fig. 9.1. — Transformarea schemei $\pi(a)$ într-o schemă stelată (b), în triunghiu multiplu (c) sau în punte (d).

nomică a schemei printr-o reducere a numărului de contacte. În acest scop se folosește curent metoda matriceală, expusă în [72, p. 162—175].

În cele ce urmează se arată că în scopul propus se poate folosi o metodă simbolică, bazată pe proprietățile conjuncției simbolurilor de marcare exprimate prin formulele (3.16)—(3.18).

În cazul transformării unei scheme π într-o schemă echivalentă stelată, trebuie satisfăcute relațiile de echivalență date de

$$f_k = y_c y_k \quad (k = 1, 2, \ldots, n). \tag{9.1}$$

Aceste relații redau analitic echivalența funcțională a celor două scheme.

Presupunem că funcțiile logice inițiale f_k, ale bazei stelei y_c și celor k ramuri ale ei y_k sînt reprezentate prin simboluri de marcare avînd același indice superior.

Pentru satisfacerea relațiilor de echivalență (9.1), avînd în vedere proprietățile conjuncției a două simboluri de marcare date de formulele (3.16)—(3.18), este necesar ca:

a) simbolurile y_k să conțină obligatoriu toate numerele de stare care definesc pe f_k, iar simbolul y_c să conțină obligatoriu cîte un reprezentant al numerelor de stare comune tuturor funcțiilor f_k;

b) în scopul formării de grupe de numere de stare adiacente cît mai mari care conduc la scheme cu număr minim de contacte, putem adăuga numere de stare arbitrare cu condiția ca acestea să nu fie comune simbolurilor lui y_c și y_k.

În cazul transformării unei scheme π într-o schemă echivalentă în triunghiu simplu (fig. 9.1*c*), relațiile de echivalență rezultă din expresiile logice aferente schemei de comutație și ele sînt

$$\begin{aligned} f_1 &= \Delta_1 \cup \Delta_2 \Delta_{12}, \\ f_2 &= \Delta_2 \cup \Delta_1 \Delta_{12}. \end{aligned} \tag{9.2}$$

În cazul cînd schema este definită prin trei funcții logice, schema echivalentă reprezintă un triunghiu multiplu, iar relațiile de echivalență se deduc prin permutarea indicilor în formula

$$f_1 = \Delta_1 \cup \Delta_2 (\Delta_{12} \cup \Delta_{13} \Delta_{23}) \cup \Delta_3 (\Delta_{13} \cup \Delta_{12} \Delta_{23}). \tag{9.3}$$

Relații de echivalență similare se pot deduce și pentru cazul cînd schema de comutație este definită prin mai mult de trei funcții

logice. Ele se deduc făcînd uz de legile algebrei booleene care se folosesc la descrierea schemelor cu contacte prin expresii logice.

Avînd în vedere proprietăţile produsului mai multor simboluri de marcare cu acelaşi indice superior, se pot stabili următoarele reguli ale transformării triunghiulare:

a) simbolurile de marcare Δ_k, luate separat, trebuie să cuprindă obligatoriu numerele de stare specifice numai funcţiei f_k. Pentru cuprinderea acestor numere de stare în grupe de numere de stare adiacente cît mai mari, se pot adăuga, în măsura în care aceasta e necesar, numere de stare noi, dar numai dintre cele cuprinse de funcţia f_k. Dacă simbolul f_k cuprinde numai numere de stare comune şi cel puţin una dintre celelalte funcţii logice, latura Δ_k poate lipsi din schema echivalentă, evident în cazul cînd prin aceasta restul celorlalte condiţii rămîn satisfăcute;

b) numerele de stare comune la două funcţii logice trebuie cuprinse cel puţin într-unul din simbolurile de marcare Δ_k specifice acestor funcţii;

c) simbolurile de marcare Δ_{jk} trebuie să cuprindă obligatoriu numerele de stare comune simbolurilor f_j şi f_k, cuprinse numai într-unul din simbolurile Δ_k sau Δ_j. Dacă astfel de numere de stare nu există, ramura Δ_{jk} lipseşte din schema echivalentă.

De asemenea puntea de legătură lipseşte şi atunci cînd laturile ce s-ar putea uni nu au nici un număr de stare comun.

În scopul formării unor grupe de numere de stare adiacente cît mai mari, se pot adăuga numere de stare arbitrare dintre cele neconţinute de funcţiile logice care definesc schema sau dintre cele comune simbolurilor Δ_j şi Δ_k.

În cazul transformării unei scheme π într-o schemă echivalentă în punte (fig. 9.1*d*), trebuie satisfăcute relaţii de echivalenţă de tipul

$$f_k = H_k F_k. \tag{9.4}$$

O astfel de problemă poate fi pusă atunci cînd funcţiile logice f_k permit scoaterea în factor a unor termeni cum este H'_k. În acest caz, problema se reduce la găsirea schemei echivalente triunghiulare pentru situaţia cînd locul funcţiilor logice f_k este luat de factorii F_k.

În fine, mai putem întîlni cazul cînd este posibilă o transformare mixtă. O astfel de situaţie se poate ivi atunci cînd, în urma transformării schemei într-una echivalentă stelată sau triunghiulară,

apare posibilitatea de a transforma din nou o parte a schemei din stea în triunghiu sau invers.

În toate cazurile cînd se cere găsirea unei scheme echivalente este necesară structurarea simbolurilor de marcare care definesc funcțiile

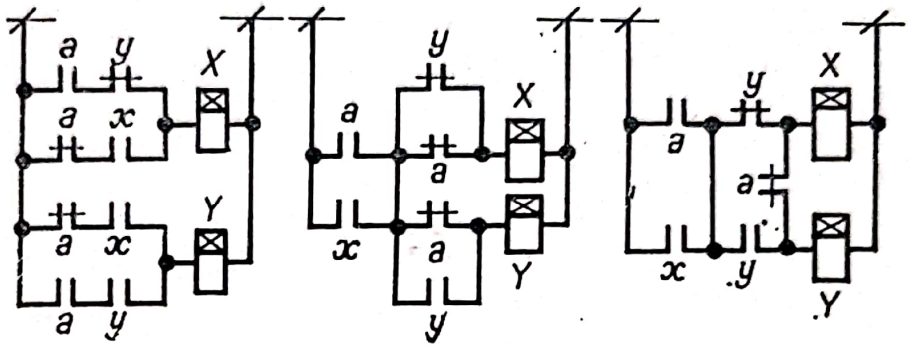


Fig. 9.2. — Scheme echivalente pentru comanda automată a avansului longitudinal la un strung revolver.

logice ale schemei de comutație. Găsirea acestor simboluri pe o cale foarte expeditivă a fost arătată în capitolul 7.

În cele ce urmează vom arăta aplicabilitatea metodei simbolice de transformare a schemelor cu contacte prin exemple luate din practica acționărilor electrice automatizate.

Exemplul 9.1. Ne propunem să găsim schemele echivalente celei date în figura 9.2*a*, schemă luată din lucrarea [6, p. 64—66].

Schema dată corespunde următoarelor funcții logice și simboluri de marcare:

$$X = a\bar{y} \cup \bar{a}x = D^{axy}_{(2,\ 3,\ 4,\ 6)},$$

$$Y = ay \cup \bar{a}x = D^{axy}_{(2,\ 3,\ 5,\ 7)}.$$

Schema echivalentă stelată, dedusă folosind regulile expuse anterior, rezultă din

$$F(X, Y) = D^{axy}_{(2,\ 3,\ 4,\ 5,\ 6,\ 7)} \cdot D^{axy}_{(0,\ 1,\ 2,\ 3,\ 4,\ 6),\ (0,\ 1,\ 2,\ 3,\ 5,\ 7)} =$$

$$= (a \cup x)\,[(\bar{a} \cup \bar{y}),\ (\bar{a} \cup y)].$$

Ultima formă de scriere a funcției logice $F(X, Y)$ evidențiază posibilitatea de a folosi în comun, pentru circuitul lui X și pentru cel al lui Y, termenul $(a \cup x)$. În același timp această formă reprezintă o exprimare mai concentrată a conținutului expresiei logice.

Numerele de stare (0) și (1) au fost adăugate arbitrar în scopul formării unor grupe de numere de stare adiacente mai mari. Găsirea implicanților primi care corespunde simbolurilor de marcare s-a făcut folosind metoda simbolică de minimizare, expusă în capitolul 6. Formarea acestor implicanți a sugerat ideea adăugării numerelor de stare amintite.

Astfel s-au format mai întîi grupările aferente funcției logice X, folosind numai numerele de stare cuprinse de simbolul de marcare dat. S-a obținut astfel

$$\begin{matrix} 1 & & 2 \\ 2 \,/\, 3 & \text{și} & 4 \,/\, 6. \end{matrix}$$

Adăugarea numerelor de stare (0) și (1) arbitrare permite formarea unor grupări mai mari, și anume

$$\begin{matrix} 1 \;\; 2 & & 2 \;\; 4 \\ 2 \,/\, 3 \,/\, 0\;1 & \text{și} & 4 \,/\, 6 \,/\, 0\;2. \end{matrix}$$

La fel s-a procedat pentru funcția logică Y și pentru simbolul de marcare aferent bazei stelei.

Observăm că schema echivalentă stelată reprezentată în figura 9.2*b* face economie de două contacte în raport cu schema π inițială. Apare evident astfel avantajul transformării efectuate. Mai mult, remarcăm posibilitatea de a transforma în triunghi schema π dată de funcțiile logice ale ramurilor stelei:

$$F'(X, Y) = (\bar{a} \cup \bar{y}),\ (\bar{a} \cup y) = D^{ay}_{(0,1)} \cup D^{ay}_{(2),(3)}.$$

Folosind regulile transformării în triunghiu, găsim

$$\Delta_1 = D^{ay}_{(0,2)} = \bar{y};\ \Delta_2 = D^{ay}_{(1,3)} = y;\ \Delta_{12} = D^{ay}_{(0,1)} = \bar{a}.$$

Schema corespunzătoare este dată în figura 8.2*c*, din care se vede că este posibilă reducerea numărului de contacte de la 8 la numai 5.

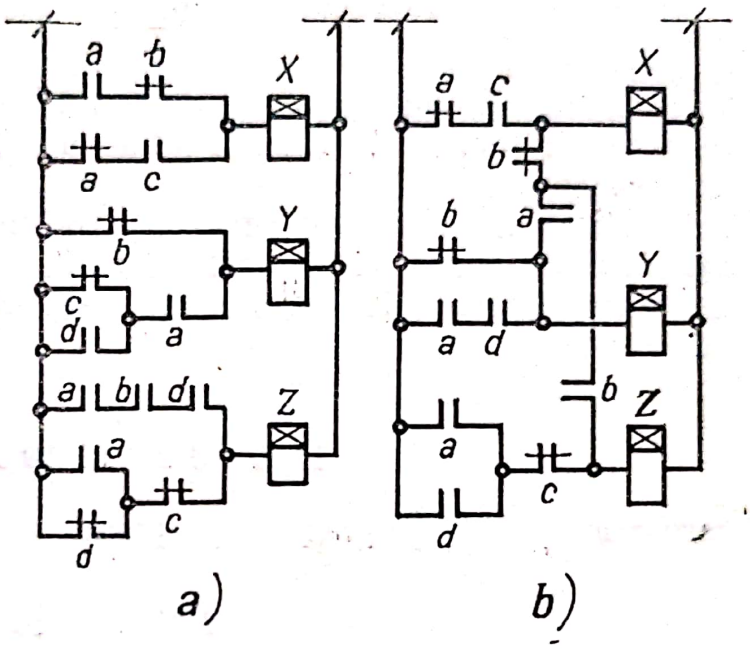


Fig. 9.3. — Schemele aferente exemplului 9.2.

Exemplul 9.2. Ne propunem să transformăm schema dată în figura 9.3. într-o schemă echivalentă triunghiulară.

Această problemă este rezolvată prin metoda matriceală în lucrarea [68, p. 173]. Cele trei funcții logice ale schemei de comutație sînt date de

$$F(X, Y, Z) = (a\bar{b} \cup \bar{a}c),\ [\bar{b} \cup a(\bar{c} \cup d)],\ [abd \cup \bar{c}(a \cup \bar{d})] =$$

$$= D^{abcd}_{(10,11)} \cup D^{abcd}_{(2,3,6,7,12,13),(0,1,2,3,12,13,14,15,17),(0,4,14,15,17)}.$$

Simbolul de marcare Δ_x trebuie să cuprindă obligatoriu numerele de stare specifice funcției logice X, adică numerele de stare (6) și (7). Cu aceste numere formăm lista de adiacențe

$$\begin{matrix} 1 & 4 \\ 6 & 7 & 2 & 3, \end{matrix}$$

care corespunde implicantului prim

$$D_{0\underline{1}1\underline{0}}^{abcd} = \bar{a}c.$$

Numerele de stare (2) şi (3) fac parte din simbolul lui X şi deci pot fi incluse în lista de adiacenţe în scopul creşterii numărului de adiacenţe.

Simbolul de marcare Δ_y trebuie să cuprindă obligatoriu numărul de stare (1), specific lui Y. Cu acesta şi cu alte numere de stare dintre cele cuprinse de simbolul Y se formează lista de adiacenţe

$$\begin{array}{c} 1 \quad 2 \quad 10 \\ 1 \,/\, 0 \,/\, 3\ 2 \,/\, 11\ 10\ 13\ 12, \end{array}$$

care corespunde implicantului prim

$$D_{\underline{0}0\underline{0}\underline{1}}^{abcd} = \bar{b}.$$

Simbolul de marcare Δ_z trebuie să cuprindă obligatoriu numărul de stare (4), specific lui Z. Cu acesta şi cu alte numere de stare dintre cele cuprinse de simbolul lui Z se formează lista de adiacenţe

$$\begin{array}{c} 4 \quad 10 \\ 4 \,/\, 0 \,/\, 14\ 10, \end{array}$$

care corespunde implicantului prim

$$D_{\underline{0}1\underline{0}0}^{abcd} = \bar{c}\bar{d}.$$

Pentru satisfacerea condiţiei b) a transformării în triunghi se impune adăugarea la simbol Δ_y şi a grupării cuprinse de lista de adiacenţe

$$\begin{array}{c} 2 \quad 4 \\ 15 \,/\, 17 \,/\, 11\ 13, \end{array}$$

care corespunde implicantului prim

$$D^{abcd}_{1\underline{1}0\underline{1}} = ad.$$

Prin această grupare se cuprind numerele de stare (15) și (17), comune funcțiilor logice Y și Z, care nu au fost cuprinse pînă acum nici în Δ_y, nici în Δ_z.

De asemenea, pentru a cuprinde și numărul de stare (14), comun funcțiilor logice Y și Z, și a numărului de stare (11), comun funcțiilor logice X și Z, se impune adăugarea unei grupări suplimentare simbolului lui Δ_z. Lista de adiacențe corespunzătoare

$$\begin{matrix} & 4 & & 1 & \\ 14 & / & 10 & / & 15 & 11 \end{matrix}$$

corespunde implicantului prim

$$D^{abcd}_{1\underline{1}0\underline{0}} = a\bar{c}.$$

Simbolul de marcare Δ_{xy} trebuie să cuprindă obligatoriu numerele de stare (10), (11), (12) și (13), care sînt comune funcțiilor logice X și Y și care nu au fost cuprinse decît în simbolul lui Δ_y. Cu ele formăm lista de adiacențe

$$\begin{matrix} & 1 & & 2 & \\ 10 & / & 11 & / & 12 & 13, \end{matrix}$$

care corespunde implicantului prim

$$D^{abcd}_{10\underline{0}\underline{0}} = a\bar{b}.$$

Simbolul de marcare Δ_{yz} trebuie să cuprindă obligatoriu numerele de stare (14) și (17), care au fost cuprinse numai într-unul din simbolurile Δ_y și Δ_z. Cu ele formăm lista de adiacențe

$$\begin{matrix} & 1 & & 2 & \\ 14 & / & 15 & / & 16 & 17, \end{matrix}$$

Scanned with OKEN Scanner

care corespunde implicantului prim

$$D^{abcd}_{11\underline{00}} = ab.$$

În fine, latura Δ_{xz} a triunghiului poate lipsi, întrucît singurele numere de stare comune funcţiilor logice X şi Z, adică numerele de stare (10) şi (11), sînt cuprinse atît de Δ_x, cît şi de Δ_z.

Rezultatele obţinute ne conduc la structurarea schemei reprezentate în figura 8.3 *b*, schemă care corespunde expresiilor logice

$$\Delta_x = \bar{a}c, \qquad \Delta_{xy} = a\bar{b},$$

$$\Delta_y = \bar{b} \cup ad, \quad \Delta_{yz} = ab.$$

$$\Delta_z = \bar{c}\bar{d} \cup a\bar{c} = \bar{c}\,(a \cup \bar{d}),$$

Avantajul transformării rezultă din reducerea numărului de contacte de la 14 la 11.

Formele minimale se obţin scriind disjuncţia implicanţilor obţinuţi din listele de adiacenţă. Dacă la întocmirea listelor de adiacenţă s-au luat în considerare numerele de stare care se referă la valorile zero ale variabilelor de ieşire, atunci forma disjunctivă obţinută poate fi uşor transformată în formă conjunctivă minimală. În acest scop se neagă de două ori rezultatul obţinut şi prin aplicare teoremei lui De Morgan se găseşte forma minimală conjunctivă, adică conjuncţia unor disjuncţii ale variabilelor de intrare.

Capitolul 10

MINIMIZAREA GLOBALĂ A EXPRESIILOR BOOLEENE

Problema minimizării structurii schemelor de comutaţie cu mai multe ieşiri a făcut obiectul multor lucrări publicate dintre care citãm lucrările [8], [9], [56], [77], [90]. Analiza acestor lucrări evidenţiază două căi posibile de minimizare: cea a minimizării individuale a funcţiilor logice aferente fiecărei variabile de ieşire şi cea a minimizării globale a sistemului de funcţii logice ce defineşte schema de comutaţie. Calea a doua conduce de regulă la soluţii optime.

În cazul structurării schemelor de comutaţie dinamică, urmînd calea minimizării individuale a funcţiilor logice se poate obţine o structură optimă prin transformarea ulterioară a schemei π obţinute, într-o schemă echivalentă stelată, triunghiulară, în punte sau mixtă.

Pentru cazul structurării schemelor de comutaţie statică, propunem următoarea metodă de minimizare, expusă în cele ce urmează.

Avînd dat simbolul de marcare unitar al schemei de comutaţie (formula 1.2), se determină mai întîi implicanţii primi specifici fiecăreia dintre funcţiile logice X_k. În acest scop se întocmesc listele de adiacenţă ce cuprind în mod obligator numerele de stare specifice funcţiilor logice X_k. Aceste liste vor cuprinde numărul maxim de numere de stare dintre cele ce definesc separat fiecare funcţie X_k.

Numerele de stare se vor bifa (prin subliniere) în şirul numerelor de stare specifice funcţiei logice respective, şir dat în forma operativă a simbolului de marcare.

După formarea acestor liste, în simbolul de marcare vor rămîne nebifate numai numerele de stare comune mai multor funcţii logice.

Considerînd variabilele de ieşire X_n în ordinea crescătoare a indicelui n şi numerele de stare ce le definesc, care au rămas nebifate, în ordinea lor crescătoare, se formează alte liste de adiacenţe, urmărind a cuprinde în acestea maximum de numere de stare

Scanned with OKEN Scanner

comune numărului maxim de funcţii logice. Aceste liste definesc implicanţii ce pot fi folosiţi în comun de către mai multe variabile de ieşire, pentru structurarea schemei de comutaţie statică.

Pe măsură ce se completează listele, se bifează numerele de stare în simbolul de marcare. Procesul de formare a listelor se continuă pînă ce toate numerele de stare au fost cuprinse, cel puţin o singură dată, într-una dintre aceste liste.

Pentru variabila de ieşire X_k, se vor considera atît implicanţii primi, specifici acestei variabile, cît şi implicanţii comuni acestei variabile şi altora.

Vom exemplifica în cele ce urmează această metodă prin trei exemple, luate din literatura curentă în scopul de a oferi posibilitatea de comparaţie a rezultatelor.

Exemplul 10.1. Se cere minimizarea globală a schemei combinaţionale de comutaţie date prin simbolul unitar de marcare:

$$F(Z_1, Z_2, Z_3) = D^{x_4x_3x_2x_1}_{\substack{(1,\ 2,\ 5^*,\ 7,\ 11,\ 12,\ 15,\ 16,\ 17),\\ (2,\ 3,\ 5^*,\ 6,\ 7,\ 10^*,\ 12,\ 14,\ 16^*,\ 17),\\ (1,\ 3,\ 5^*,\ 6,\ 7,\ 10^*,\ 11,\ 12,\ 14,\ 16,\ 17).}}$$

Indicele (*), specifică valorile binare indiferente ale variabilelor de ieşire.

Analiza numerelor de stare cuprinse de forma operativă a simbolului de marcare dat prin enunţ, evidenţiază faptul că numai funcţia logică Z_1 are specific numărul de stare (15). Cu acesta se formează lista de adiacenţe:

Pentru Z_1

$$\begin{array}{c} \overset{2}{\cdot}\quad 10 \\ 15\ /\ 17\ /\ 5\ \ 7 \end{array}$$

ce corespunde implicantului prim

$$D^{x_4x_3x_2x_1}_{1\ 1\ 0\ 1} = x_1x_3.$$

Luînd în considerare numerele de stare comune, în ordinea specificată prin regulile date mai sus, se formează listele de adiacențe :

$$\text{Pentru } Z_1 \text{ și } Z_3: \quad \overset{10}{1} / 11 \rightarrow D_{\underline{0}\,0\,0\,1}^{x_4x_3x_2x_1} = x_1\bar{x}_2\bar{x}_3.$$

$$Z \text{ și } Z_2: \quad \overset{10}{2} / 12 \rightarrow D_{\underline{0}\,0\,1\,0}^{x_4x_3x_2x_1} = \bar{x}_1x_2\bar{x}_3.$$

$$Z_1 \text{ și } Z_3: \quad \overset{1}{16} / 17 \rightarrow D_{1\,1\,1\,\underline{0}}^{x_4x_3x_2x_1} = x_2x_3x_4.$$

$$Z_2 \text{ și } Z_3: \quad \overset{4}{3} / 7 \rightarrow D_{0\,\underline{0}\,1\,1}^{x_4x_3x_2x_1} = x_1x_2\bar{x}_4.$$

$$\overset{10}{6} / \overset{1}{16} / 7\ 1 \rightarrow D_{\underline{0}\,1\,1\,\underline{0}}^{x_4x_3x_2x_1} = x_2x_3.$$

$$\overset{2}{14} / \overset{4}{16} / 10\ 12 \rightarrow D_{1\,\underline{1}\,\underline{0}\,0}^{x_4x_3x_2x_1} = \bar{x}_1x_4.$$

Se obțin astfel expresiile minimale globale

$$Z_1 = x_1x_3 \cup x_1\bar{x}_2\bar{x}_3 \cup \bar{x}_1x_2\bar{x}_3 \cup x_2x_3x_4.$$

$$Z_2 = \bar{x}_1x_2\bar{x}_3 \cup x_1x_2x_4 \cup x_2x_3 \cup \bar{x}_1x_4.$$

$$Z_3 = x_1\bar{x}_2\bar{x}_3 \cup x_2x_3x_4 \cup x_1x_2\bar{x}_4 \cup x_2x_3 \cup \bar{x}_1x_4 =$$

$$= x_1\bar{x}_2\bar{x}_3 \cup x_1x_2\bar{x}_4 \cup x_2x_3 \cup \bar{x}_1x_4.$$

Aceste rezultate coincid cu cele date în [77].

Unica variantă posibilă, în aplicarea regulilor acestei metode ar fi condus la folosirea implicantului : $\bar{x}_1x_2x_4$, în locul lui : $x_2x_3x_4$, această variantă rezultînd din formarea listei de adiacențe

$$\overset{4}{16} / 12 \text{ în loc de : } \overset{1}{16} / 17.$$

În ambele cazuri se obţine un raport de structură optim, dat de:

$$K_s = \frac{4 + 7 + 6}{4 + 18 + (12 + 3)} = \frac{17}{37} \frac{\text{module}}{\text{intrări}}.$$

Exemplul 10.2. Se cere minimizarea globală a schemei combinaţionale de comutaţie date prin simbolul de marcare:

$$F(X, Y, Z) = D^{x_1x_2x_3x_4}_{\substack{(6,\ 7,\ 10,\ 11,\ 12,\ 13,\ 14,\ 15,\ 16,\ 17),\\ (1,\ 3,\ 6,\ 7,\ 10,\ 11,\ 12,\ 13),\\ (1,\ 3,\ 6,\ 7,\ 10,\ 14,\ 15,\ 16,\ 17).}}$$

Folosind metoda de minimizare globală expusă mai sus, se formează listele de adiacenţe.

Pentru: X, Y şi Z: $\overset{1}{6} / 7 \rightarrow D^{x_1x_2x_3x_4}_{0\ 1\ 1\ \underline{0}} = \bar{x}_1x_2x_3.$

$10 \rightarrow D^{x_1x_2x_3x_4}_{1\ 0\ 0\ 0} = x_1\bar{x}_2\bar{x}_3\bar{x}_4.$

X şi Y: $\overset{2}{11} / \overset{1}{13} / 10\ 12 \rightarrow D^{x_1x_2x_3x_4}_{1\ 0\ \underline{0}\ \underline{1}} = x_1\bar{x}_2.$

X şi Z: $\overset{1}{14} / \overset{2}{15} / 16\ 17 \rightarrow D^{x_1x_2x_3x_4}_{1\ 1\ \underline{0}\ \underline{0}} = x_1x_2.$

Y şi Z: $\overset{2}{1} / 3 \rightarrow D^{x_1x_2x_3x_4}_{0\ 0\ \underline{0}\ 1} = \bar{x}_1\bar{x}_2x_4.$

Se obţin astfel expresiile logice global minimale.

$$X = \bar{x}_1x_2x_3 \cup x_1\bar{x}_2 \cup x_1x_2.$$

$$Y = \bar{x}_1x_2x_3 \cup x_1\bar{x}_2 \cup \bar{x}_1\bar{x}_2x_4.$$

$$Z = \bar{x}_1x_2x_3 \cup x_1\bar{x}_2\bar{x}_3\bar{x}_4 \cup x_1x_2 \cup \bar{x}_1\bar{x}_2x_4.$$

Scanned with OKEN Scanner

Rezultatul obținut diferă de cel dat în [90], unde dintr-o eroare s-au omis unii termeni, astfel încît de exemplu funcția X, în forma minimală dată în lucrare, nu satisface stările active (11), (12) și (13).

Schema modulară UNILOG, ce corespunde acestor formule, are un raport de structură dat de

$$K_s = \frac{4+5+6}{4+14+(10+3)} = \frac{15}{31} \frac{\text{module}}{\text{intrări}};$$

Dacă minimizarea globală s-ar fi făcut prin luarea în considerare a stărilor pasive (ce corespund la valorile 0 a variabilelor de ieșire) s-ar obține soluția:

$$\overline{X} = \bar{x}_1 x_2 \bar{x}_3 \cup \bar{x}_1 \bar{x}_2.$$

$$\overline{Y} = \bar{x}_1 x_2 \bar{x}_3 \cup \bar{x}_1 \bar{x}_2 \bar{x}_4 \cup x_1 x_2.$$

$$\overline{Z} = \bar{x}_1 x_2 \bar{x}_3 \cup \bar{x}_1 \bar{x}_2 \bar{x}_4 \cup x_1 \bar{x}_2 x_4 \cup x_1 \bar{x}_2 x_3.$$

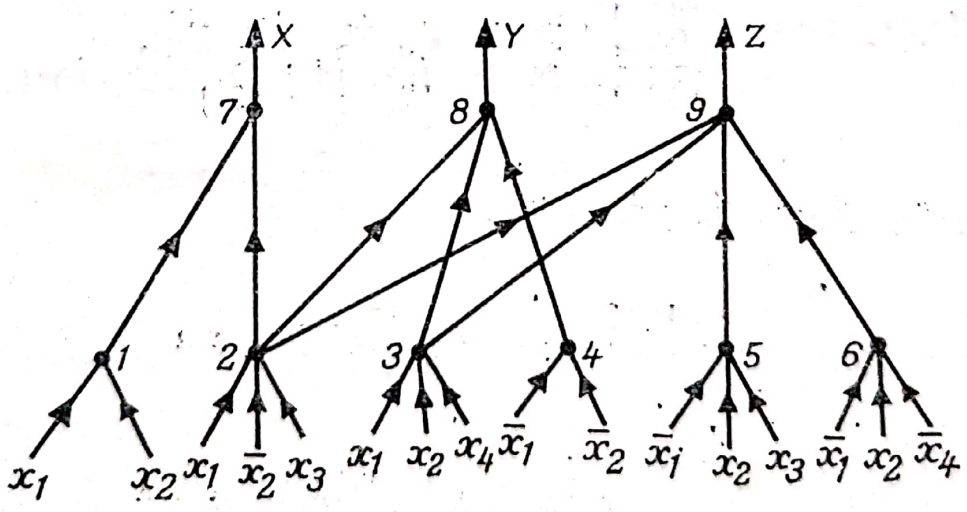


Fig. 10.1. — Schema modulară UNILOG pentru exemplul 10.2.

Această soluție corespunde schemei modulare UNILOG reprezentate în figura 10.1 sub forma unui graf orientat.

Scanned with OKEN Scanner

Schema corespunde unui raport de structură.

$$K_s = \frac{4 + 6 + 3}{4 + 10 + (9)} = \frac{13}{29} \frac{\text{module}}{\text{intrări}}$$

deci, mai avantajos decît în primul caz. Raportul de structură calculat astfel ţine seama şi de modulele necesare pentru negarea logică a variabilelor de intrare.

Exemplul 10.3. Se cere minimizarea globală a sistemului de funcţii logice dat de:

$$F(z_1, z_2, z_3) = D^{x_1x_2x_3x_4}_{\substack{(4,\ 5,\ 7,\ 10,\ 11,\ 12^*,\ 13,\ 16,\ 17),\\ (1^*,\ 3,\ 4,\ 5,\ 6,\ 7^*,\ 12^*,\ 14,\ 16,\ 17),\\ (1^*,\ 3,\ 5,\ 6,\ 7,\ 10,\ 11,\ 12^*,\ 14,\ 16,\ 17).}}$$

Alegînd calea de minimizare prin considerarea stărilor pasive ale schemei, exprimate prin simbolul de marcare:

$$F(z_1, z_2, z_3) = D^{x_1x_2x_3x_4}_{\substack{(0^0,\ 1^0,\ 2^0,\ 3^0,\ 6^0,\ 12^*,\ 14^0\ 15^0),\\ (0^0,\ 1^*,\ 2^0,\ 7^*,\ 10^0,\ 11^0,\ 12^*,\ 13^0,\ 15^0),\\ (0^0,\ 1^*,\ 2^0,\ 4^0,\ 12^*,\ 13^0,\ 15^0).}}$$

se formează listele de adiacente:

Pentru z_1: $\overset{1\quad 2}{1\ /\ 0\ /\ 3\ \ 2} \rightarrow D^{x_1x_2x_3x_4}_{0\ 0\ \underline{0}\ \underline{1}} = \bar{x}_1\bar{x}_2.$

$$\overset{4}{6\ /\ 2} \rightarrow D^{x_1x_2x_3x_4}_{0\ \underline{1}\ 1\ 0} = \bar{x}_1x_3\bar{x}_4.$$

$$\overset{1}{14\ /\ 15} \rightarrow D^{x_1x_2x_3x_4}_{1\ 1\ 0\ \underline{0}} = x_1x_2\bar{x}_3.$$

Scanned with OKEN Scanner

Pentru z_2: $\overset{1}{10} / \overset{2}{11} / 12\ 13 \rightarrow D^{x_1x_2x_3x_4}_{1\ 0\ \underline{0}\ \underline{0}} = x_1\bar{x}_2.$

Pentru z_3: $\overset{4}{4} / 0 \rightarrow D^{x_1x_2x_3x_4}_{0\ \underline{1}\ 0\ 0} = \bar{x}_1\bar{x}_3\bar{x}_4.$

Pentru z_1, z_2 și z_3: $\overset{2}{0} / 2 \rightarrow D^{x_1x_2x_3x_4}_{0\ 0\ \underline{0}\ 0} = \bar{x}_1\bar{x}_2\bar{x}_4.$

$15 \rightarrow D^{x_1x_2x_3x_4}_{1\ 1\ 0\ 1} = x_1x_2\bar{x}_3x_4.$

Pentru z_2 și z_3: $\overset{1}{13} / 12 \rightarrow D^{x_1x_2x_3x_4}_{1\ 0\ 1\ \underline{1}} = x_1\bar{x}_2x_3.$

Rezultă deci expresiile global minimale:

$$z_1 = \bar{x}_1\bar{x}_2 \cup \bar{x}_1x_3\bar{x}_4 \cup x_1x_2\bar{x}_3.$$

$$z_2 = x_1\bar{x}_2 \cup \bar{x}_1\bar{x}_2\bar{x}_4 \cup x_1x_2\bar{x}_3x_4.$$

$$z_3 = \bar{x}_1\bar{x}_3\bar{x}_4 \cup \bar{x}_1\bar{x}_2\bar{x}_4 \cup x_1x_2\bar{x}_3x_4 \cup x_1\bar{x}_2x_3.$$

Presupunînd că la intrarea schemei dispunem atît de variabilele de intrare cît și de negațiile lor, rezultă un raport de structură al schemei modulare UNILOG, dat de:

$$K_s = \frac{8+3}{23+10} = \frac{11}{33}\ \frac{\text{module}}{\text{intrări}}.$$

Se obține astfel un raport de structură mai avantajos din punctul de vedere al numărului de intrări la module decît cel dat

în lucrarea [9], unde se obține:

$$K_s = \frac{11}{36} \frac{\text{module}}{\text{intrări}}.$$

Rezultă deci concluzia că aplicarea metodei propuse în acest capitol în scopul minimizării globale a unui sistem de funcții logice conduce la rezultate comparabile cu cele obținute prin alte metode, dar pe o cale mai expeditivă. Metoda se pretează bine pentru calculul manual și se poate aplica la sisteme de funcții logice cu mai mult de șase variabile de intrare, cu dificultăți mai mici decît cele specifice în acest caz altor metode.

Scanned with OKEN Scanner

Capitolul 11

METODĂ SIMBOLICĂ DE SINTEZĂ A SCHEMELOR DE COMUTAȚIE

În general, prin sinteză se înțelege operația de concepere (proiectare) a circuitelor sistemelor automate.

În cazul sintezei schemelor de comutație se pleacă de la specificațiile cerute schemei și se cere în final schema de comutație care le poate realiza practic. Rezultă deci că operația de sinteză este inversă operației de analiză.

11.1. SINTEZA SCHEMELOR DE COMANDĂ AUTOMATĂ A MOTOARELOR ELECTRICE

Schemele de comutație folosite pentru comanda automată a motoarelor electrice controlează operațiile de pornire, oprire, frînare și inversare a sensului de rotație a acestor motoare. Alături de aceste funcții principale, schemele de comutație folosite pentru comanda automată a motoarelor electrice mai îndeplinesc încă o serie de funcții, care pot fi considerate secundare. Dintre acestea fac parte, de exemplu, funcția de semnalizare optică sau acustică, funcția de telecomandă și de protecție, precum și diversele blocări folosite în cadrul acestor scheme.

Punctul de plecare în procesul de sinteză al unei scheme de comutație îl constituie enunțarea specificațiilor pe care trebuie să le îndeplinească aceasta. Ele reprezintă datele inițiale de proiectare a schemei de comutație. Plecînd de la aceste date, se stabilesc sub formă analitică sau grafică secvențele de lucru și stările de blocare care intervin în funcționarea schemei.

Folosind informațiile cuprinse de secvențele de lucru și de stările de blocare ale schemei de comutație, se pot stabili simbolurile de marcare asociate diferitelor variabile de ieșire ale ei sau, în

Scanned with OKEN Scanner

cazul unor scheme de comutaţie relativ simple, simbolul de marcare unic al acesteia.

Procesul de sinteză se continuă cu explicitarea simbolurilor de marcare într-o formă minimală adică cu găsirea expresiilor logice minimale ale schemei. În acest scop se va folosi în această lucrare metoda simbolică de minimizare.

Formulele minimale obţinute astfel permit structurarea schemelor de comutaţie dinamică folosind dispozitive cu contacte, sau schemele de comutaţie statică, care folosesc elemente fără contacte.

Economicitatea realizării practice a schemei de comutaţie dinamică se va aprecia funcţie de valoarea unui coeficient „de structură", notat prin K_d şi definit prin raportul dintre numărul de relee sau de contactoare folosite în schemă şi numărul de contacte necesare îndeplinirii funcţiilor schemei, contacte care aparţin fie releelor sau contactoarelor, fie unor dispozitive de comandă, cum ar fi butoanele, întrerupătoarele, limitatoarele etc. :

$$K_d = \frac{n_a}{n_c}\left[\frac{\text{număr de aparate}}{\text{număr de contacte}}\right]. \qquad (11.1)$$

Avîndu-se în vedere că în ţara noastră s-a generalizat folosirea elementelor de comutaţie statică UNILOG, economicitatea realizării schemelor modulare de comutaţie statică se va aprecia funcţie de valoarea coeficientului de structură K_s, dat de raportul

$$K_s = \frac{n_v + n_m + n_d}{n_i}\left[\frac{\text{număr de module „nici"}}{\text{număr total de intrări}}\right], \qquad (11.2)$$

unde n_v reprezintă numărul variabilelor de intrare nenegate, n_m numărul implicanţilor care definesc schema de comutaţie statică, n_d numărul disjuncţilor negate necesare pentru realizarea practică a funcţiei logice şi n_i numărul total de intrări ale schemei de comutaţie.

Pentru definirea acestui coeficient s-a avut în vedere ca numărătorul să reprezinte numărul modulelor „nici" necesare schemei, iar numitorul numărul tuturor intrărilor folosite de acestea. Astfel, luînd în considerare forma disjunctivă minimală a expresiilor logice, pentru fiecare variabilă de intrare nenegată care intră într-un implicant (conjuncţie) este necesar cîte un modul „nici" care realizează negaţia logică a variabilei aplicate la intrarea modulului „nici"

Scanned with OKEN Scanner

prin care se efectuează conjuncţia logică. Realizarea practică a unui implicant presupune folosirea unui modul „nici" la intrarea căruia se aplică variabilele care intră în conjuncţia respectivă.

Facem remarca că prin modul de definire a coeficientului K_i se ţine seama de posibilitatea folosirii în comun a aceluiaşi implicant de către mai multe variabile de ieşire.

Pentru realizarea disjuncţiei logice negate a implicanţilor conţinuţi de formulele logice, este necesar un modul „nici", iar pentru realizarea numai a disjuncţiei logice sînt necesare două module „nici".

Prin n_i se înţelege suma intrărilor la modulele „nici" care realizează conjuncţiile variabilelor şi a celor de la intrarea modulului care realizează disjuncţia negată sau nenegată.

În cazul disjuncţiei nenegate trebuie luată în considerare o intrare în plus, necesară pentru realizarea funcţiei logice „sau" cu ajutorul modulelor „nici".

În vederea realizării cît mai economice a unei scheme de comutaţie dinamică, se pune problema transformării schemei într-una echivalentă stelată, triunghiulară, în punte sau mixtă, iar în cazul schemelor de comutaţie statică se pune problema folosirii în comun de către mai multe variabile de ieşire a unora dintre termenii formulelor care definesc aceste variabile. În acest fel se pot obţine coeficienţi de structură mai avantajoşi din punctul de vedere al posibilităţilor de realizare practică a schemei de comutaţie.

În cadrul procesului de sinteză a unei scheme de comutaţie trebuie abordată şi problema economicităţii realizării acesteia, ceea ce se poate face luînd în considerare valorile coeficienţilor de structură definiţi mai sus. Pentru reprezentarea ca desen a schemelor de comutaţie se foloseşte curent simbolistica recomandată de normele interne sau internaţionale.

În cazul schemelor de comutaţie UNILOG, considerăm foarte potrivită şi avantajoasă o reprezentare sub formă de graf orientat, reprezentare folosită în cadrul exemplelor de sinteză ce vor fi tratate în cele ce urmează. Într-o astfel de reprezentare, modulele „nici" sînt folosite în exclusivitate pentru realizarea practică a schemei şi, ca atare, se pot reprezenta sub formă de puncte, fiecare punct reprezentînd în acelaşi timp un nod al grafului.

Conexiunile dintre aceste module reprezintă ramurile orientate ale grafului, iar sensul figurat pe ele reprezintă sensul de transmitere a semnalelor binare.

Reprezentarea schemelor de comutaţie statică UNILOG sub formă de graf orientat oferă avantajul simplităţii de reprezentare şi al uşurinţei de urmărire a conexiunilor intermodule, în scopul realizării efective a lor sau al modelării schemei de comutaţie pe un simulator logic.

Particularităţile folosirii metodei simbolice de sinteză a schemelor de comutaţie în diverse cazuri concrete vor face obiectul exemplelor care urmează.

Exemplul 11.1 Să se facă sinteza schemei de comandă automată a pornirii unui motor asincron prin cuplare directă la reţeaua trifazată de alimentare. Comenzile de pornire şi de oprire sînt date prin intermediul a două butoane cu revenire, P şi O.

Variabilele primare prin care se iniţiază procesul de pornire şi, respectiv, de oprire a motorului M sînt variabilele P şi O, asociate contactelor celor două butoane de comandă.

Pentru cuplarea la reţeaua de alimentare este suficient un singur contactor L. Pentru memorarea comenzilor date (butoanele sînt cu revenire) se va folosi variabila secundară l, asociată variabilei de ieşire L.

Precizînd ordinea variabilelor din indicele superior al simbolului de marcare POl, se pot codifica stările secvenţelor de lucru ale schemei, întocmindu-se în prealabil diagrama de faze din figura 11.1.

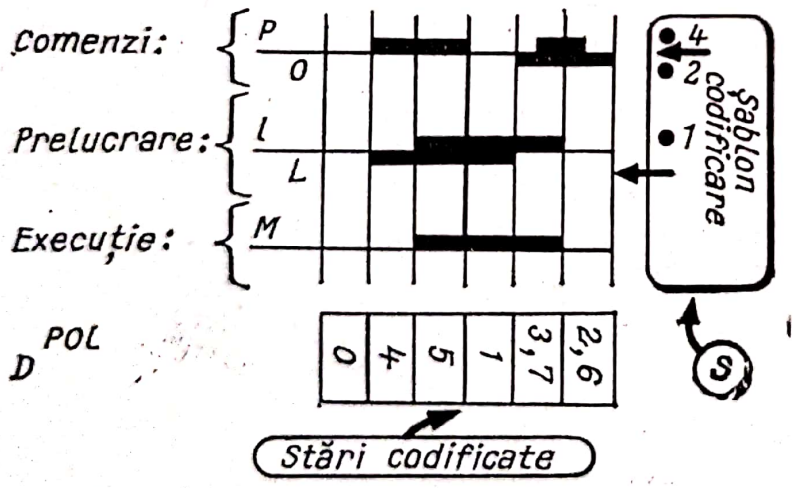


Fig. 11.1. — Diagrama de faze pentru exemplul 11.1.

Astfel, plecînd din starea (0), redată în figura 11.1, prin apăsarea butonului P se trece în starea (4), în care devine excitată bobina contactcrului L. Comutarea contactelor acestui contactor

($l = 1$) trece schema în starea (5), apoi revenirea butonului P în starea normală trece schema în starea (1).

Intervenţia butonului O determină oprirea motorului M, trecînd schema prin stările (3), (2) în starea (0). Comanda prin butonul O este prioritară, lucru exprimat prin stările suplimentare (7) şi (6).

Iniţial se întocmeşte diagrama de faze, ţinîndu-se seama de condiţiile care se impun funcţionării schemei de comandă automată. Apoi, folosind un şablon făcut din hîrtie sau din carton, notat prin S în figura 9.1, se înseamnă pe acesta poziţia variabilelor. Deplasînd apoi şablonul în lungul diagramei de faze, se pot uşor codifica stările secvenţelor de lucru ale schemei, numerele de stare determinate astfel scriindu-se ca în figura 9.1.

Diagrama de faze codificată permite scrierea uşoară a simbolurilor de marcare aferente variabilelor de ieşire ale schemei. În cazul nostru avem

$$F(L, M) = D^{POl}_{(1, 4, 5), (1, 3, 5, 7)} = \overline{O}(P \cup l), l.$$

Expresiile logice, aferente acestor simboluri rezultă imediat prin formarea grupelor de adiacenţe cerute de metoda simbolică de minimizare.

Schema cu contacte care corespunde formulelor logice deduse este dată în figura 11.2 *a* şi este identică cu cea dedusă, făcînd sinteza prin metoda folosirii directe a formulei de interpolare a lui Lagrange [13, p. 230].

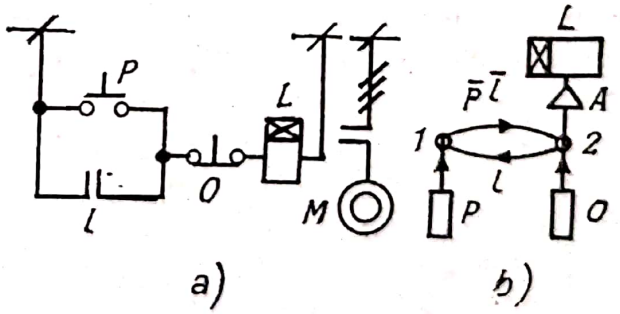


Fig. 11.2. — Comanda automată a pornirii unui motor asincron.

Pentru deducerea schemei de comutaţie statică realizată cu module „nici", se transformă mai întîi expresia booleeană astfel încît să conţină numai operatori „$\top$". În cazul nostru avem

$$L = \overline{\overline{O}(P \cup l)} = \overline{O \cup (P \cup l)} = O \top (P \top l).$$

După cum se observă din figura 11.2*b*, schema sintetizată se poate realiza cu numai două module „⊤". Prin reprezentarea sub formă de graf orientat se evidenţiază bucla de reacţie folosită în schemă, buclă specifică elementelor cu memorie.

În cazul nostru, ansamblul celor două module „⊤" formează de fapt un circuit basculant bistabil cu două intrări, şi anume intrarea de înscriere a memoriei prin variabila P şi cea de ştergere a ei, prin variabila O, precum şi unica ieşire folosită.

Amplificatorul A serveşte pentru ridicarea nivelului puterii semnalului de ieşire, astfel încît acesta să fie suficient de puternic pentru excitarea bobinei contactorului L. Remarcăm că pentru a fi o schemă de comutaţie statică în întregimea ei, ar trebui eliminat şi contactorul L. Rolul lui poate fi îndeplinit în acest caz de grupări antiparalel de tiristoare, înseriate pe cele trei faze de alimentare a motorului asincron trifazat M.

Pentru simplificarea expunerii şi uşurinţa înţelegerii funcţionării schemelor sintetizate, la ieşirile acestora vom folosi întotdeauna numai relee sau contactoare, subînţelegînd posibilitatea înlocuirii lor prin tiristoare.

Coeficienţii de structură ai schemelor sintetizate sînt

$$K_a = \frac{1}{6}\ \text{ap./cont.}\ \text{şi}\ K_s = \frac{2}{4}\ \text{mod./intr.}$$

Exemplul 11.2. Să se facă sinteza schemei de comandă automată a pornirii unui motor asincron cu rotor scurtcircuitat cu limitarea curentului de pornire în circuitul statorului şi protecţia maximală de curent.

Comenzile sînt date prin butoanele P şi O, iar pentru controlul scurtcircuitării rezistoarelor înseriate (reactoarelor) cu înfăşurările statorului se foloseşte un contact cu temporizare mecanică controlat de un contactor auxiliar A.

Cuplarea normală la reţeaua de alimentare se face prin contactele unui contactor L.

Considerînd variabilele primare P, O şi R_M şi variabilele secundare a, a_t şi l, rezultă indicele simbolului de marcare $POR_M\, a_t\, al$.

Ţinînd seama de condiţiile ce trebuie îndeplinite în funcţionarea schemei de comandă automată, se stabileşte diagrama de

faze din figura 11.3. În această diagramă este specificată și prioritatea comenzilor date prin releul maximal R_M sau prin butonul de oprire O. De asemenea, prin M_p și M_n s-a înțeles mersul motorului în regim de pornire și în regim normal.

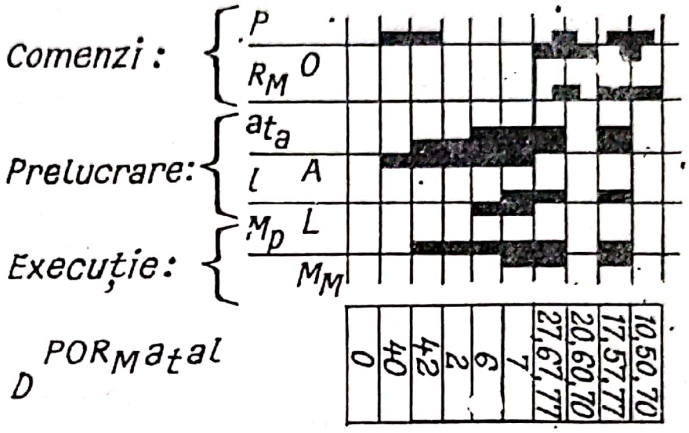


Fig. 11.3. — Diagrama de faze pentru exemplul 11.2.

Codificînd stările descrise de diagrama de faze, deducem următorul simbol de marcare al schemei de comutație:

$$F(A, L, M) = D^{POR_M a_t al}_{1:(2,6,7,40,42);\ 0:(0,10,17,20,27,50,57,60,67,70,77)}$$
$$1:(6,7);\ 0:(0,2,10,17,20,27,40,42,50,57,60,67,70,77)$$
$$M_p: 1:(2,6,7,17,27,42,57,67,77);\ 0:(0,10,20,40,50,60,70)$$
$$M_n: 1:(7,17,27,57,77);\ 0:(0,2,6,10,20,40,42,50,60,70).$$

Funcția logică pentru circuitul contactorului A rezultă din

$$\begin{matrix}1 & 4 & & 40\\ 2 & 3 & 6 & 7 & 42 & 43 & 46 & 47\end{matrix}$$
$$D^{POR_M a_t a\, l}_{\underline{0}\,0\,0\ \underline{0}\,1\underline{0}} = \overline{O}\overline{R}_M\, a,$$

$$\begin{matrix}1 & 2 & & 4\\ 40 & 41 & 42 & 43 & 44 & 45 & 46 & 47\end{matrix}$$
$$D^{POR_M a_t al}_{1\,0\,0\ \ \underline{0}\,\underline{0}\underline{0}} = P\overline{O}\overline{R}_M.$$

Rezultă

$$A = \overline{O}\,\overline{R}_{M}(P \cup a).$$

Pentru contactorul L avem

$$\begin{array}{l} 1\quad 2\quad\ 40 \\ 6\,/\,7\,/\,4\ 3\,/\,46\ 47\ 44\ 43 \end{array}$$

$$D^{PO R_{M} a_{t} al}_{\underline{0}\,0\ 0\ \ 1\,\underline{1}\underline{0}} = \overline{O}\,\overline{R}_{M}\, a_{t}.$$

Rezultă

$$L = \overline{O}\,\overline{R}_{M}\, a_{t}.$$

Pentru circuitul de pornire al motorului şi cel de regim normal avem

$$\begin{array}{l} \phantom{20\,\frac{2}{22}\,/\,}1\quad 4\quad\ 10 \\ 20\,\frac{2}{22}\,/\,3\,/\,6\ 7\,/\,12\ 13\ 16\ 17 \\ 40\,\frac{}{42} \\ 62 \end{array}$$

$$M_{p} = D^{PO R_{M} a_{t} al}_{\underline{0}\,\underline{0}\,\underline{0}\ \ \underline{0}\,\underline{1}\underline{0}} = a.$$

şi

$$\begin{array}{l} \phantom{20\,\frac{7}{27}\,/\,}2\quad 4\quad\ 10 \\ 20\,\frac{7}{27}\,/\,5\,/\,3\ 1\,/\,17\ 15\ 13\ 11 \\ 40\,\frac{}{47} \\ 67 \end{array}$$

$$M_{n} = D^{PO R_{M} a_{t} al}_{\underline{0}\,\underline{0}\,\underline{0}\ \ \underline{1}\,\underline{1}\underline{1}} = l.$$

Schemele cu și fără contacte deduse din aceste formule sînt date în figurile 11.4*a* și 11.4*b*.

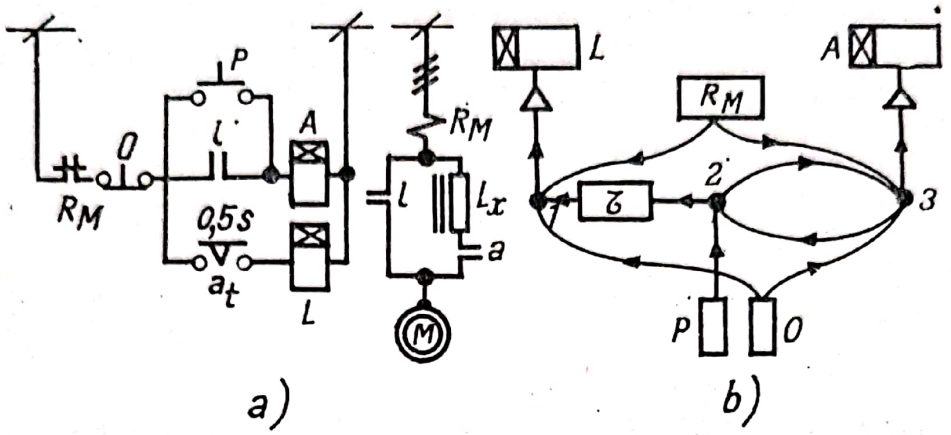


Fig. 11.4. — Comanda automată a pornirii unui motor cu limitarea curentului în stator.

Exemplul 11.3. Să se facă sinteza schemei de comandă automată reversibilă cu interblocarea comenzii a unui motor asincron cu rotor scurtcircuitat. Comenzile de pornire sînt date prin acționarea cîte unui buton pentru fiecare sens de rotație P_s și P_d, iar oprirea este comandată prin acționarea butonului de oprire O. Butoanele sînt cu revenire.

În scopul simplificării procesului de sinteză, în indicele simbolului de marcare aferent schemei, $P_d P_s ds$, nu am cuprins variabila primară O. De funcția acestei variabile se poate ține seama numai la scrierea formulei finale a schemei, întrucît butonul O are rol de întrerupător general.

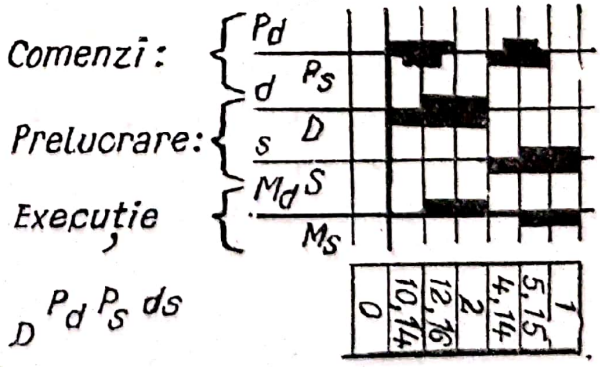


Fig. 11.5. — Diagrama de faze pentru exemplul 11.3.

Secvențele de lucru din figura 11.5, deduse din specificațiile schemei sintetizate, țin seama de prioritatea primei comenzi date

printr-unul din butoanele P_d sau P_s, asigurînd astfel interblocarea cerută prin enunţul problemei de sinteză.

Din diagrama de faze rezultă următorul raport de stare al schemei care realizează funcţiunile cerute prin enunţ:

$$F(D, S, M_d, M_s) =$$

$$= \overline{O}\left[D^{P_d P_s ds}_{(3^*, 6^*, 7^*, 11^*, 13^*, 14^*, 17^*)} \cup D^{P_d P_s ds}_{\substack{(2, 10, 12, 16),\\ (1, 4, 5, 15),\\ (2, 12, 16),\\ (1, 5, 15)}}\right].$$

Facem remarca că, dacă în urma codificării stărilor un număr de stare apare atît pentru valoarea 0 a variabilei de ieşire, cît şi pentru valoarea 1 a acesteia, înseamnă că el corespunde unei stări indiferente. Acesta este, de exemplu, cazul stării (14), care apare atît pentru $S = 0$, cît şi pentru $S = 1$.

Prin minimizarea efectuată, folosind metoda simbolică, se ajunge la

$$\begin{matrix} 1 & 4 & & 10 \\ 2 & 3 & 6 & 7 & 12 & 13 & 16 & 17 \end{matrix}$$

$$D^{P_d P_s ds}_{\underline{0}\ \underline{0}\ \underline{10}} = d,$$

$$\begin{matrix} 4 & & 2 \\ 10 & 14 & 12 & 16 \end{matrix}$$

$$D^{P_d P_s ds}_{1\ \underline{0}\ \underline{00}} = P_d \overline{s}.$$

Rezultă

$$D = (P_d \overline{s} \cup d)\,\overline{O},$$

$$\begin{matrix} 4 & 10 & & 2 \\ 1 & 5 & 11 & 15 & 3 & 7 & 13 & 17 \end{matrix}$$

$$D^{P_d P_s ds}_{\underline{0}\ \underline{0}\ \underline{01}} = s,$$

$$\begin{matrix} 10 & & 1 \\ 4 & 14 & 5 & 15 \end{matrix}$$

$$D^{P_d P_s ds}_{\underline{0}\ 1\ \underline{00}} = P_s \overline{d}$$

Rezultă

$$S = (P_s \overline{d} \cup s)\,\overline{O},$$

$$\begin{matrix} 10 & 4 & & 1 \\ 2 & 12 & 6\ 16 & 3\ 13\ 7\ 17 \end{matrix}$$

$$M_d = D_{\underline{0}}^{P_d P_s ds} = d,$$

$$\begin{matrix} 4 & 10 & & 2 \\ 1 & 5 & 11\ 15 & 3\ 7\ 13\ 17 \end{matrix}$$

$$M_s = D_{\underline{0}}^{P_d P_s ds} = s.$$

Expresiile logice ale funcțiilor D și S se pot scrie și sub forma

$$D = \overline{s}\,(P_d \cup d)\,\overline{O} \quad \text{și} \quad S = \overline{d}\,(P_s \cup s)\,\overline{O},$$

avînd în vedere că la deducerea lor nu era obligatorie folosirea diferenței de adiacență 1, respectiv 2. Sub ultima formă apar mai evident formele structurale ale celor două memorii, date în figurile 11.6*a* și 11.6*b*.

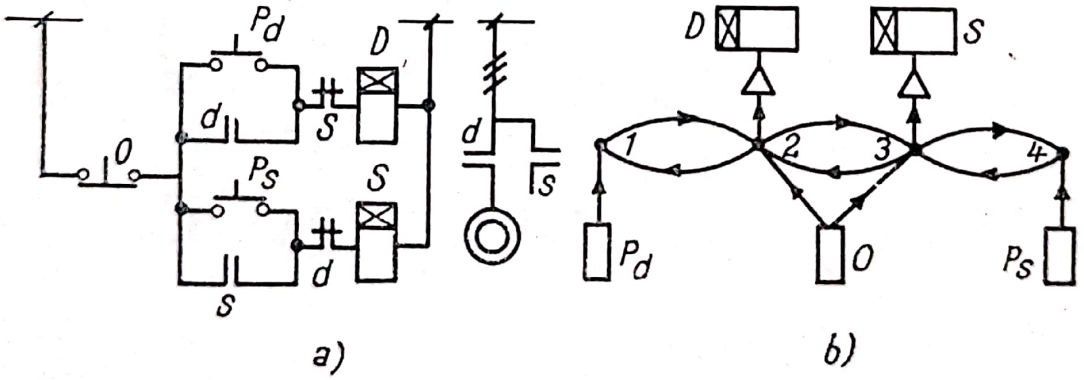


Fig. 11.6. — Comanda automată reversibilă cu interblocare a unui electromotor.

Prin nefolosirea diferențelor de adiacență 1 pentru primul implicant a lui D, respectiv 2 pentru primul implicant a lui S acești implicanți se scriu respectiv

$$d(\overline{s}) \quad \text{și} \quad s(\overline{d}),$$

prin paranteze punîndu-se în evidenţă variabilele redondante, prin folosirea cărora adesea se poate ajunge la o formă mai convenabilă a structurii schemei de comutaţie.

Sinteza acestei scheme prin folosirea directă a formulei de interpolare a lui Lagrange este dată în [13, p. 230]. Acolo se structurează numai schema cu contacte.

Exemplul 11.4. Să se facă sinteza schemei de comandă automată fără interblocarea comenzii pentru un electromotor. Comenzile de pornire sînt date prin acţionarea a cîte unui buton pentru fiecare sens de rotaţie (P_d şi P_s), iar oprirea este comandată prin butonul de oprire O, care joacă rol şi de întrerupător de avarie. Butoanele sînt cu revenire.

La fel ca în exemplul precedent, funcţia butonului O va fi considerată abia în formulele finale.

Secvenţele de lucru date în figura 11.7 ţin seama de prioritatea acţionării unuia dintre butoanele de comandă.

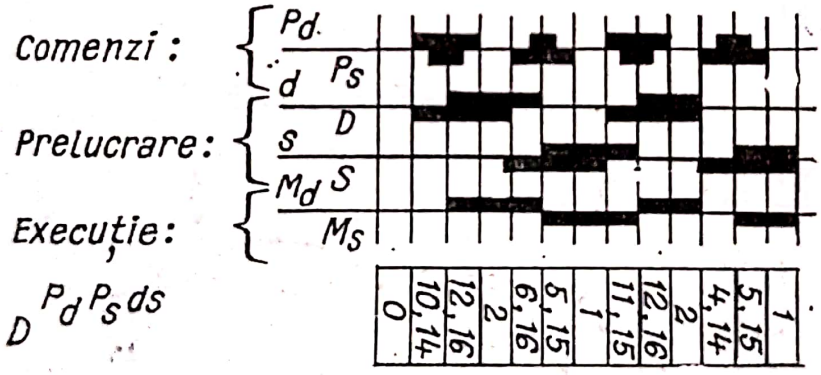


Fig. 11.7. — Diagrama de faze pentru exemplul 11.4.

Din diagrama de faze rezultă simbolul de marcare al schemei :

$$F(D, S, M_d, M_s) = \overline{O}\left[D^{P_d P_s ds}_{(3^*,\ 13^*,\ 17^*)} \cup D^{P_d P_s ds}_{\substack{(2,10,11,12,14^*,15^*,16^*)\\(1,\ 4,\ 5,\ 6,\ 14^*,\ 15^*,\ 16^*)\\(2,\ 6,\ 12,\ 16),\ (1,\ 5,\ 11,\ 15)}}\right].$$

Prin exprimarea în forma minimală a acestui simbol, se ajunge la

$$\left.\begin{array}{ll} \overset{10}{2} \,/\, 12 \,/\, \overset{1}{3}\ 13 & D_{\underline{0}}^{P_d P_s ds} = \overline{P}_s d \\ \\ \overset{1}{2\underset{12}{\overset{10}{}}} \,/\, \overset{4}{11} \,/\, 14\ 15 & D_{1\ \underline{0}\ \underline{00}}^{P_d P_s ds} = P_d(\overline{P}_s) \end{array}\right\} D = (P_s d \cup P_d)\,\overline{O},$$

$$\left.\begin{array}{ll} \overset{4}{1} \,/\, \overset{2}{5} \,/\, 3\ 7 & D_{0\ \underline{0}\ \underline{01}}^{P_d P_s ds} = \overline{P}_d s. \\ \\ 10\underset{14}{\overset{1}{\overset{4}{}}} \,/\, \overset{2}{5} \,/\, 6\ 7 & D_{\underline{0}\ 1\ \underline{00}}^{P_d P_s ds} = P_s(\overline{P}_d) \end{array}\right\} S = (\overline{P}_d s \cup P_s)\,\overline{O},$$

$$\overset{4\ 10}{2} \,/\, 6 \,/\, 12\ 16 \,/\, \overset{1}{3}\ 7\ 13\ 17 \qquad M_d = D_{\underline{0}\ \underline{0}\ \underline{10}}^{P_d P_s ds} = d,$$

$$\overset{4\ 10}{1} \,/\, 5 \,/\, 11\ 15 \,/\, \overset{2}{3}\ 7\ 13\ 17 \qquad M_s = D_{\underline{0}\ \underline{1}\ \underline{00}}^{P_d P_s ds} = s.$$

Expresiile logice ale funcțiilor D și S se mai pot scrie și sub forma

$$D = \overline{P}_s(P_d \cup d)\,\overline{O} \quad \text{și} \quad S = \overline{P}_d(P_s \cup s)\,\overline{O},$$

avînd în vedere neobligativitatea folosirii tuturor adiacențelor posibile, adică a luării în considerare și a variabilelor redundante.

Prin ultimele formule s-au pus mai bine în evidență cele două memorii ale schemei.

Schemele cu și fără contacte, structurate după formulele deduse, s-au dat în figura 11.8.

Exemplul 11.5. Să se facă sinteza schemei de comandă automată a pornirii reversibile a unui motor serie de c.c. cu două înfășurări de excitație (fig. 11.9).

Scanned with OKEN Scanner

Comenzile de pornire-oprire se dau cu ajutorul a trei butoane O, P_s și P_d, iar cuplarea la rețeaua de alimentare se va realiza cu ajutorul a două contactoare D și S. Se impune interblocarea comenzilor date prin P_d și P_s și prioritatea comenzii de oprire.

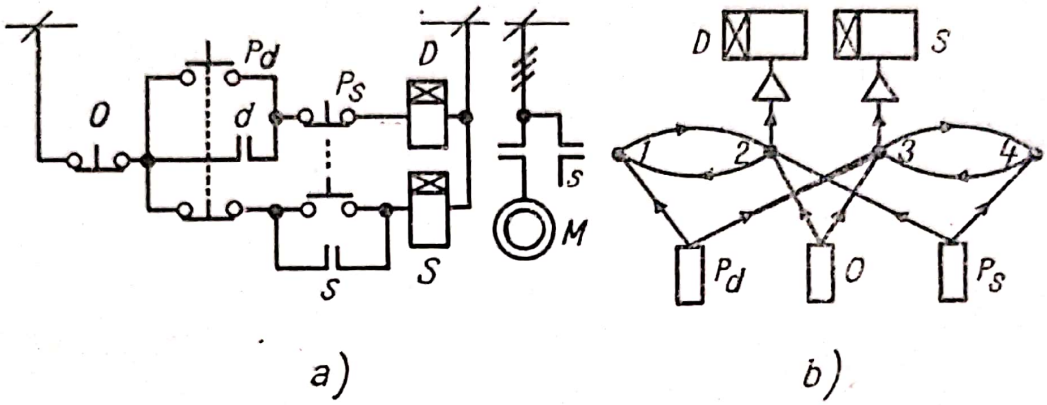


Fig. 11.8. — Comanda automată reversibilă fără interblocare a unui electromotor.

La darea comenzii de oprire din mersul spre dreapta, indusul se șuntează prin înfășurarea de excitație folosită în mersul spre stînga, iar la darea comenzii de oprire din mersul spre stînga, indusul se șuntează prin înfășurarea de excitație folosită în mersul spre dreapta. În acest fel poate fi asigurată o frînare rapidă a motorului.

Examinînd schema din figura 11.9, observăm că întrucît starea de repaus a motorului succedă regimului de frînare care corespunde situației cînd $F_s = 1$ sau $F_d = 1$, vor exista două stări

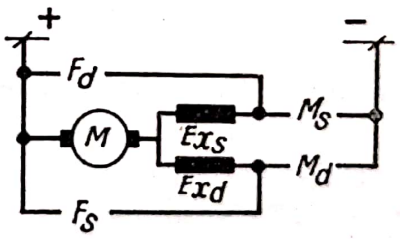


Fig. 11.9. — Schema aferentă exemplului 11.5.

posibile de repaus. Pentru discriminarea acestor două stări se impune folosirea unei variabile auxiliare de ieșire F, pentru care se asociază variabila secundară f. Astfel, o stare de repaus se va caracteriza prin $F = 0$, iar a doua stare de repaus prin $F = 1$.

În diagrama de faze din figura 11.10, secvențele S_1 și S_3 redau procesul de pornire spre dreapta și de oprire din sensul de rotație spre dreapta, iar secvențele S_2 și S_4 redau procesul de pornire spre stînga și de oprire din sensul de rotație spre stînga, plecînd de la cele două stări posibile de repaus.

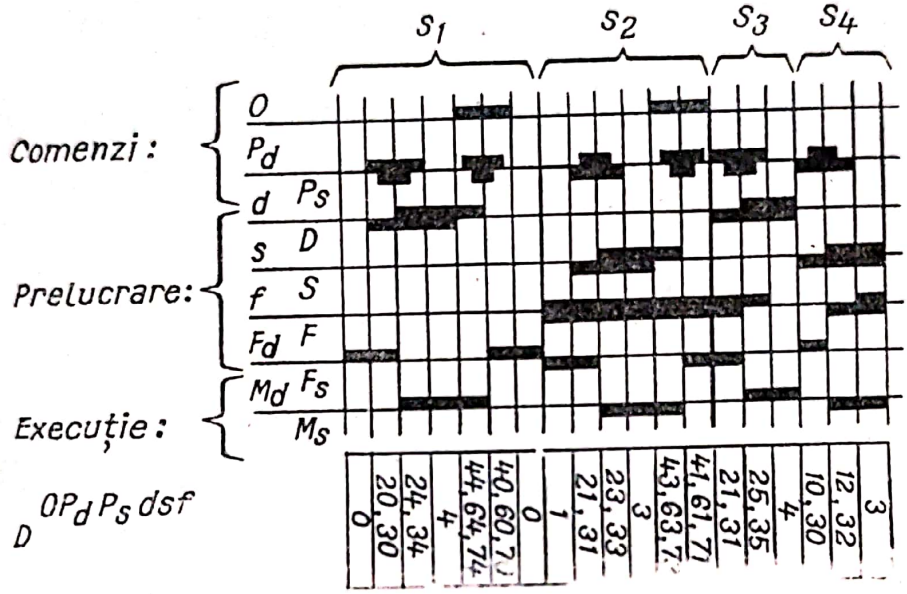


Fig. 11.10. — Diagrama de faze pentru exemplul 11.5.

Din diagrama stărilor (de faze) rezultă simbolurile de marcare care, explicitate în formă minimală, ne dau funcțiile logice căutate. Astfel se obține :

— pentru F :

$$F = D^{OP_d P_s dsf}_{1:(1,3,12,21,23,31,32,33,41,43,61,63,71,73),\ 0:(0,4,10,20,24,25,30,34,35,44,60,64,70,74),}$$

$$40\,\frac{1}{41}\,/\,\frac{2}{3}\,/\,\frac{20}{21\ 23}\,/\,\frac{10}{11\ 13\ 31\ 33},$$

$$D^{OP_d P_s dsf}_{\underline{00}\ \underline{0}\ \underline{001}} = f\bar{d},$$

$$\begin{array}{l} \quad\ 20\ \ 10\ \ \ 40 \\ 1\ \overline{12}/\ 32\ /\ 2\ 22\ /\ 52\ 72\ 42\ 62, \\ 4\ \overline{13} \\ \quad\ \overline{16} \\ \quad\ \cdot\cdot \end{array}$$

$$D_{\underline{00}\ \underline{1}\ \underline{010}}^{OP_aP_sdsf} = s(\overline{d});$$

în paranteză este trecută variabila redundantă, adăugată numai pentru a putea pune expresia logică finală sub forma

$$F = \overline{d}(f \cup s);$$

— pentru D:

$$D = D_{1:(4,\ 20,\ 21^*,\ 24,\ 25,\ 30^*,\ 31^*,\ 35),}^{OP_aP_sdsf}$$
$$0:(0,\ 1,\ 3,\ 10,\ 12,\ 23,\ 32,\ 33,\ 40,\ 41,\ 43,$$
$$44,\ 60,\ 61,\ 63,\ 64,\ 70,\ 71,\ 73,\ 74),$$

$$\begin{array}{l} \quad\ 20\ \ \ 1\ \ \ \ 10 \\ 2\ \overline{4}/\ 24\ /\ 5\ 25\ /\ 14\ 34\ 15\ 35, \\ \quad\ \overline{6} \end{array}$$

$$D_{\underline{00}\ \underline{0}\ \underline{100}}^{OP_aP_sdsf} = \overline{O}d(\overline{s}),$$

$$\begin{array}{l} \quad\ 10\ \ \ 1\ \ \ \ 4 \\ 20\ /\ 30\ /\ 21\ 31\ /\ 24\ 34\ 25\ 35, \end{array}$$

$$D_{01\ \underline{0}\ \underline{000}}^{OP_aP_sdsf} = \overline{O}P_a\overline{s};$$

rezultă deci

$$D = \overline{O}\overline{s}(P_a \cup d).$$

— pentru S:

$$S = D_{1:(3,\ 10,\ 12,\ 21^*,\ 23,\ 30^*,\ 31^*,\ 32,\ 33),}^{OP_aP_sdsf}$$
$$0:(0,\ 1,\ 4,\ 20,\ 24,\ 25,\ 34,\ 35,\ 40,\ 41,$$
$$43,\ 44,\ 60,\ 61,\ 63,\ 64,\ 70,\ 71,\ 73,\ 74),$$

Scanned with OKEN Scanner

$$1\ \frac{3}{2}\ /\ 23\ /\ 13\ 33\ /\ 7\ 27\ 17\ 37,$$

$$D_{\underline{00}\ \underline{0}\ \underline{011}}^{OP_dP_sdsf} = \overline{O}s(\overline{a}),$$

$$10\ /\ 12\ /\ 11\ 13\ /\ 30\ 32\ 31\ 33,$$

$$D_{\underline{00}\ 1\ \underline{000}}^{OP_dP_sdsf} = \overline{O}P_s\overline{d};$$

rezultă deci

$$S = \overline{O}\,\overline{d}(P_s \cup s);$$

— pentru F_d:

$$F_d = D_{1:(0,\,10,\,20,\,30,\,40,\,60,\,70),\ 0:(1,\,3,\,4,\,12,\,21,\,23,\,24,\,25,\,31,\,32,\,33,\,34,\,35,\,41,\,43,\,44,\,61,\,63,\,64,\,71,\,73,\,74),}^{OP_dP_sdsf}$$

$$0\ /\ 10\ /\ 20\ 30\ /\ 40\ 50\ 60\ 70,$$

$$D_{\underline{00}\ \underline{0}\ 000}^{OP_dP_sdsf} = \overline{d}\,\overline{s}\,\overline{f};$$

rezultă deci

$$F_d = \overline{d}\,\overline{s}\,\overline{f};$$

— pentru F_s:

$$F_s = D_{1:(1,\,21,\,31,\,41,\,61,\,71),\ 0:(0,\,3,\,4,\,10,\,12,\,20,\,23,\,24,\,25,\,30,\,32,\,33,\,34,\,35,\,40,\,43,\,44,\,60,\,63,\,64,\,70,\,73,\,74),}^{OP_dP_sdsf}$$

$$1\ /\ 21\ /\ 11\ 31\ /\ 41\ 61\ 51\ 71,$$

$$D_{\underline{00}\ 0\ 001}^{OP_dP_sdsf} = \overline{d}\,\overline{s}f;$$

rezultă deci

$$F_s = \bar{d}\bar{s}f.$$

Funcțiile logice M_s și M_d rezultă direct din diagrama de faze. Din aceasta se observă că

$$M_d = d \text{ și } M_s = s.$$

Schema cu contacte structurată, folosind funcțiile logice deduse mai sus, este reprezentată în figura 11.11 *a*.

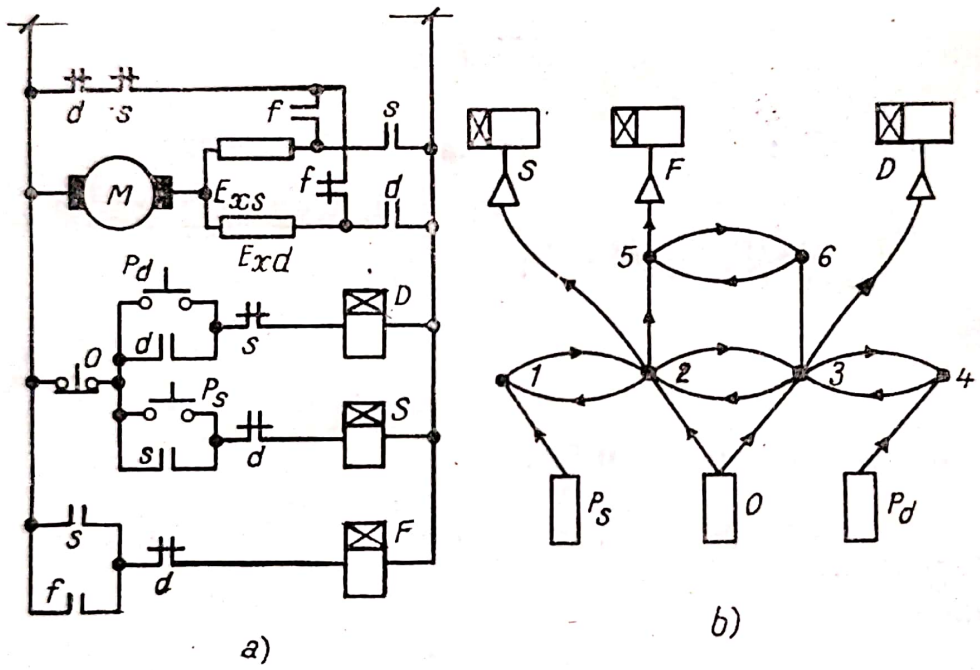


Fig. 11.11. — Comanda automată a pornirii reversibile și frînării unui motor serie de c.c.

Funcțiile logice ale variabilelor de ieșire asociate celor trei contactoare folosite în schemă evidențiază existența a trei circuite de memorie. Ele apar mai evident în schema modulară UNILOG, dată în figura 11.11 *b*, schemă care se referă numai la comanda celor trei contactoare.

Exemplul 11.6. Să se facă sinteza schemei de comandă automată a pornirii nereversibile a unui motor de c.c. cu scurtcircuitarea temporizată în două trepte a reostatului de pornire. Comenzile de

pornire și oprire sînt date prin acționarea butoanelor cu revenire P și O, iar temporizarea este asigurată de două relee de timp R_1 și R_2.

Pentru controlul scurtcircuitării celor două trepte ale reostatului de pornire se impune folosirea a două contactoare auxiliare A și B.

Avînd în vedere că starea contactorului L și a primului releu de timp R_1 depinde numai de variabilele P, O și l, iar starea celui de-al doilea releu de timp R_2 și a contactoarelor auxiliare A și B depinde numai de contactele r_1, r_2 și a, vom considera două simboluri de marcare și vom codifica diagrama de faze din figura 11.12., corespunzător acestora.

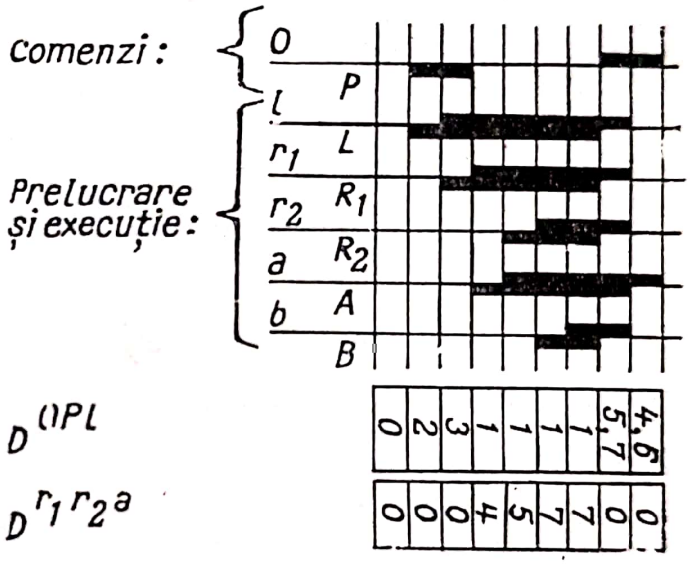


Fig. 11.12. — Diagrama de faze pentru exemplul 11.6.

Din această diagramă rezultă

$$L = D^{OPl}_{(1,\,2,\,3)} = \overline{O}\,(P \cup l),$$

$$R_1 = D^{OPl}_{(1,\,3)} = \overline{O}l,$$

$$R_2 = D^{r_1\,r_2\,a}_{(5,\,7,\,1^*,\,2^*,\,3^*,\,6^*)} = a,$$

$$A = D^{r_1\,r_2\,a}_{(4,\,5,\,7,\,1^*,\,2^*,\,3^*,\,6^*)} = r_1,$$

$$B = D^{r_1\,r_2\,a}_{(7,\,1^*,\,2^*,\,3^*,\,6^*)} = r_2.$$

Scanned with OKEN Scanner

Rezultă schemele cu şi fără contacte, date în figurile 11.13*a* şi 11.13*b*. Remarcăm că în schema fără contacte temporizările sînt asigurate cu ajutorul elementelor modulare „timp".

Schema cu contacte care face obiectul acestui exemplu este sintetizată prin metoda folosirii directe a formulei de interpolare a lui Lagrange în lucrarea [13, p. 231].

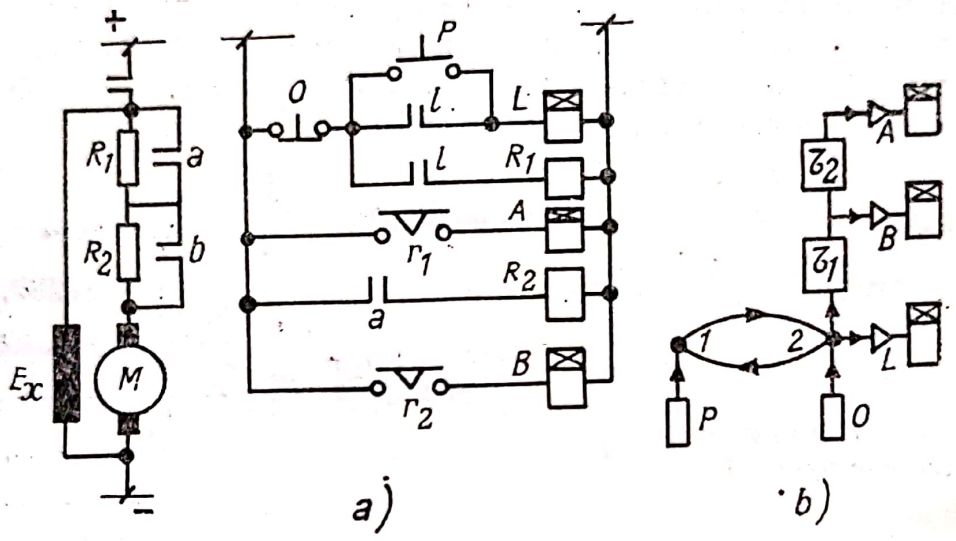


Fig. 11.13. — Comanda automată a pornirii funcţie de timp a unui motor de c.c.

Exemplul 11.7. Să se facă sinteza schemei de comandă automată reversibilă cu frînare prin cuplare inversă şi scurtcircuitarea temporizată în două trepte a reostatului de pornire a unui motor de c.c. derivaţie.

Controlul sensului de rotaţie şi a stării de repaus se face cu ajutorul a două relee conectate la bornele indusului prin două elemente redresoare.

Comenzile de pornire-oprire se fac prin trei butoane cu revenire P_a, P_s şi O.

Diagrama de faze, trasată în figura 11.14 evidenţiază secvenţele de lucru ale schemei. Realizarea funcţiilor cerute schemei impune folosirea contactoarelor D şi S, prin care se asigură cuplarea motorului la reţea pentru cele două sensuri de rotaţie, a contactoarelor C_i, A_1 şi A_2, prin care se controlează rezistoarele din circuitul indusului motorului, a releelor de timp R_1 şi R_2 şi a celor de sens R_d şi R_s.

Trasarea diagramei de faze s-a făcut avîndu-se în vedere specificațiile schemei și posibilitățile de comutare ale releelor și a contactoarelor folosite. Se impune o atenție deosebită la stabilirea acestei diagrame, evitîndu-se trecerile între stări neadiacente.

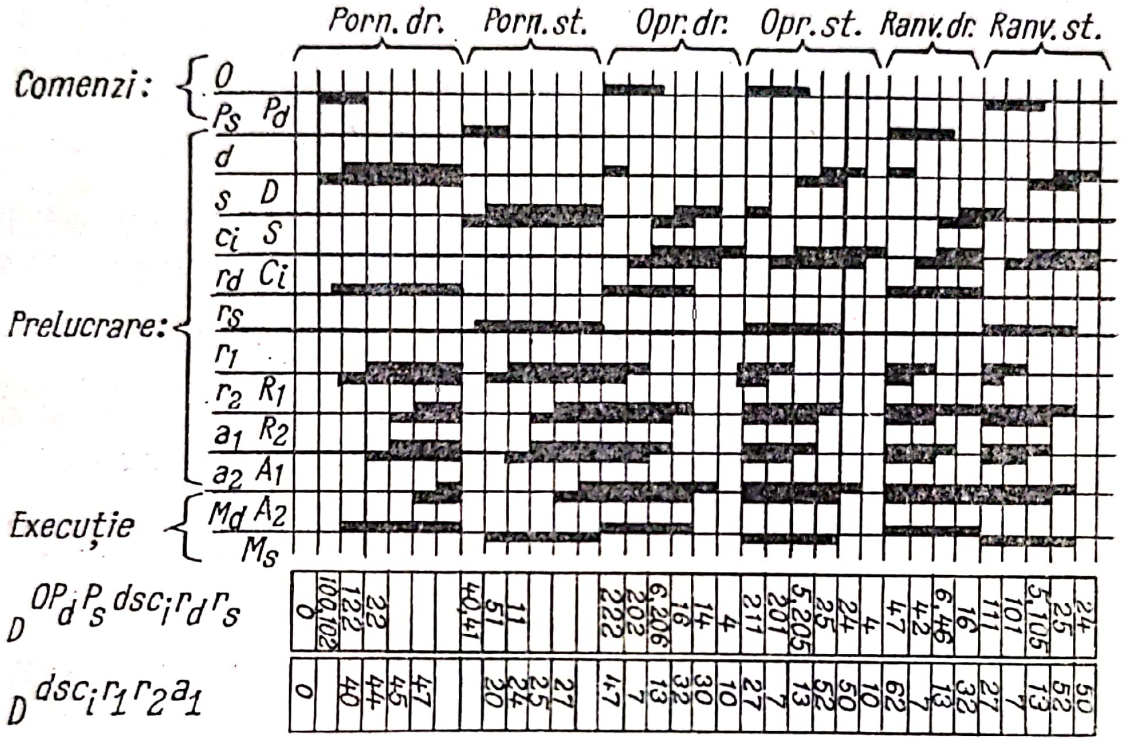


Fig. 11.14. — Diagrama de faze pentru exemplul 11.7.

Secvențele de lucru ale schemei corespund la pornirea spre dreapta, oprirea din sensul de rotație spre dreapta și ranversarea sensului de rotație plecînd de la rotația spre dreapta. Celelalte trei secvențe, care corespund cazului cînd motorul se rotește spre stînga se trasează ușor prin analogie cu primele secvențe, deduse pe cale logică.

Trebuie să subliniem că rezultatul sintezei depinde total de reflectarea în diagrama de faze a specificațiilor schemei. Trecerile neadiacente între stări cuprinse de același simbol de marcare duce la apariția hazardului de curse în funcționarea schemei. Pentru a le evita nu trebuie să admitem ca schimbarea valorii binare a unei variabile de intrare să modifice simultan valorile binare a mai multor variabile de ieșire, conținute de același simbol de marcare, ci cel

mult numai a uneia singure. Este vorba de variabilele de ieşire care au ataşate şi variabile secundare.

O variabilă de intrare poate însă modifica simultan valorile binare a mai multor variabile de ieşire, care însă nu au asociate şi variabile secundare cuprinse în acelaşi simbol de marcare.

Funcţionarea schemei de comutaţie poate fi descrisă de cele două simboluri de marcare indicate în diagrama de faze. Prin această divizare se simplifică procesul de sinteză. Ea este posibilă atunci cînd variabilele de ieşire ale schemei nu depind de toate variabilele de intrare, ci numai de o parte a acestora.

În cazul nostru se poate considera că variabilele de ieşire D, S şi C_i depind de un grup de variabile de intrare, în timp ce restul celorlalte variabile de ieşire depind de alt grup de variabile. Aceste dependenţe se pot deduce din diagrama de faze.

Codificarea stărilor pentru cele două simboluri de marcare se face folosind şablonul de codare, ca în exemplul 11.1. Se deduc astfel numerele de stare înscrise pe diagrama de faze din figura 11. 14.

Pentru funcţia logică D, se obţine

$$D = D^{OP_a\ P_s\ dsc_i\ r_a\ r_s}_{1:(5,\ 22,\ 25,\ 105,\ 102,\ 100,\ 122,\ 205),\ 0:(0,\ 4,\ 6,\ 11,\ 14,\ 16,\ 24,\ 40,\ 41,\ 42,46,\ 51,\ 62,\ 101,\ 111,\ 201,\ 202,\ 206,\ 211,\ 222),}$$

$$10\frac{5}{15}\ /\ 25\ /\ 105\ 125\ /\ 7\ 27\ 107\ 127,$$
$$40\frac{15}{45}$$
$$200\frac{25}{205}$$
$$\ldots$$

$$D^{OP_a\ P_s\ dsc_i\ r_a\ r_s}_{0\ \underline{0}\ \ \underline{0}\ \ 0\ \ \underline{001}\ \underline{0}\ 1} = c_i\ r_s,$$

$$10\ \frac{22}{32}\ /\ 122\ /\ 2\ 102\ /\ 23\ 123\ 3\ 103,$$

$$D^{OP_a\ P_s\ dsc_i\ r_a\ r_s}_{0\ \underline{0}\ \ 0\ \ \underline{100}\ 1\ \underline{0}} = \overline{OP}_s\ \overline{c}_i\ r_a.$$

$$\begin{array}{r} \quad 200 \quad 40 \qquad 20 \\ 10\ \underline{100}\ /\ 300\ /\ 140\ 340\ /\ 120\ 320\ 160\ 360, \\ 4\ \underline{110} \\ \quad 104 \\ 2\ \underline{114} \\ \quad 102 \\ \quad \ldots \end{array}$$

$$D^{OP_d\ P_s\ dsc_i\ r_d\ r_s}_{\underline{0}\ 1\ \ \underline{0}\ \ \underline{000}\ \underline{0}\ \ 0} = P_d\,\bar{r}_s.$$

Rezultă deci

$$D = \overline{O}\,\overline{P}_s\,\bar{c}_i\,r_d \cup P_d\,\bar{r}_s \cup c_i\,r_s.$$

Simetria comenzilor ne mai permite să scriem

$$S = \overline{O}\,\overline{P}_d\,\bar{c}_i\,r_s \cup P_s\,\bar{r}_d \cup c_i\,r_d.$$

Pentru funcţia logică a contactorului de cuplare inversă, C_i, deducem

$$C_i = D^{OP_d\ P_s\ dsc_i\ r_d\ r_s}$$
$$1 : (5,\ 6,\ 14,\ 16,\ 24,\ 25,\ 42,\ 46,\ 101,\ 105,\ 201,\ 202,\ 205,\ 206),$$
$$0 : (0,\ 4,\ 11,\ 22,\ 40,\ 41,\ 51,\ 62,\ 100,\ 102,\ 111,\ 122,\ 211,\ 222),$$

$$\begin{array}{r} \quad 20 \quad 100 \qquad 200 \\ 4\ \underline{5}\ /\ 25\ /\ 105\ 125\ /\ 205\ 225\ 305\ 325, \\ 2\ \underline{1} \\ \quad 7 \\ \quad \ldots \end{array}$$

$$D^{OP_d\ P_s\ dsc_i\ r_d\ r_s}_{\underline{0}\ \underline{0}\ \ 0\ \ \underline{001}\ \underline{0}\ 1} = \overline{P}_s\,\bar{s}r_s,$$

$$\begin{array}{r} \quad 10 \quad 40 \qquad 200 \\ 4\ \underline{6}\ /\ 16\ /\ 46\ 56\ /\ 206\ 216\ 246\ 256, \\ 1\ \underline{2} \\ \quad 7 \\ \quad \cdot \end{array}$$

$$D^{OP_d\ P_s\ dsc_i\ r_d\ r_s}_{\underline{0}\ 0\ \ \underline{0}\ \ \underline{001}\ 1\ \underline{0}} = \overline{P}_d\,\bar{d}r_d,$$

$$\begin{array}{lc} & \overset{1}{} \quad \overset{2}{} \quad\quad \overset{20}{} \\ 40 & \underline{14} \;/\; 15 \;/\; 16\ 17 \;/\; 34\ 35\ 36\ 37, \\ 100 & \underline{54} \\ & 114 \\ 200 & \underline{154} \\ & 214 \\ & \ldots \end{array}$$

$$D^{OP_d\ P_s\ dsc_i\ r_d\ r_s}_{\underline{0}\ \underline{0}\ \ \underline{0}\ \ \underline{0}11\ \underline{0}\ \ \underline{0}} = sc_i,$$

$$\begin{array}{lc} & \overset{1}{} \quad \overset{2}{} \quad\quad \overset{10}{} \\ & \underline{24} \;/\; 25 \;/\; 26\ 27 \;/\; 34\ 35\ 36\ 37 \\ 40 & \underline{64} \\ 100 & 124 \\ 200 & \underline{164} \\ & 224 \\ & \ldots \end{array}$$

$$D^{OP_d\ P_s\ dsc_i\ r_d\ r_s}_{\underline{0}\ \underline{0}\ \ \underline{0}\ \ 1\underline{0}1\ \underline{0}\ \ \underline{0}} = dc_i.$$

Rezultă deci

$$C_i = \overline{P}_s\, \overline{s}r_s \cup \overline{P}_d\, \overline{d}r_d \cup c_i\,(s \cup d).$$

Pentru R_1 avem

$$R_1 = D^{dsc_i\ r_1\ r_2\ a_1}_{\begin{array}{l}1 : (20,\ 24,\ 25,\ 27,\ 40,\ 44,\ 45,\ 47),\\ 0 : (0,\ 7,\ 10,\ 13,\ 30,\ 32,\ 50,\ 51),\end{array}}$$

$$\begin{array}{lc} & \overset{4}{} \quad \overset{1}{} \quad\quad \overset{2}{} \\ 40 & \underline{20} \;/\; 24 \;/\; 21\ 25 \;/\; 22\ 26\ 23\ 27, \\ & \underline{60} \end{array}$$

$$D^{dsc_i\ r_1\ r_2\ a_1}_{\underline{0}10\ \underline{0}\ \ \underline{0}\ \ \underline{0}} = s\overline{c}_i,$$

$$20\ \frac{40}{60}\ \overset{1}{/}\ 41\ \overset{2}{/}\ 42\ 43\ \overset{4}{/}\ 44\ 45\ 46\ 47,$$

$$D^{dsc_i\ r_1\ r_2\ a_1}_{1\underline{00}\ \underline{0}\ \underline{0}\ \underline{0}} = d\bar{c}_i .$$

Rezultă deci

$$R_1 = \bar{c}_i\ (d \cup s).$$

Pentru R_2 avem

$$R_2 = D^{dsc_i\ r_1\ r_2\ a_1}_{1:\ (7,\ 13,\ 25,\ 27,\ 45,\ 47);\ 0:\ (0,\ 10,\ 20,\ 24,\ 30,\ 32,\ 40,\ 44,\ 50,\ 52),}$$

$$2\ \frac{7}{4\ \frac{5}{3}}\ \overset{20}{/}\ 27\ \overset{40}{/}\ 47\ 67,$$

$$10\ \frac{1}{17}$$

$$\cdot\cdot$$

$$D^{dsc_i\ r_1\ r_2\ a_1}_{\underline{000}\ \underline{1}\ \underline{1}\ 1} = a_1 .$$

Rezultă

$$R_1 = a_1 .$$

Pentru A_1, avem

$$A_1 = D^{dsc_i\ r_1\ r_2\ a_1}_{1:\ (7,\ 24,\ 25,\ 27,\ 44,\ 45,\ 47),\ 0:\ (0,\ 10,\ 13,\ 20,\ 30,\ 32,\ 40,\ 50,\ 52),}$$

$$\begin{array}{l} \quad\ 10 \quad 1 \quad\ 2 \\ 20\ \underline{7}\ /\ 17\ /\ 6\ 16\ /\ 5\ 15\ 4\ 14, \\ 40\ \underline{27} \\ \quad\ \underline{47} \\ \quad\ \cdot\cdot \end{array}$$

$$D^{dsc_i\ r_1\ r_2\ a_1}_{\underline{000}\ 1\ \underline{1}\ \underline{1}} = r_1.$$

Rezultă

$$A_1 = r_1.$$

Pentru A_2 avem

$$A_2 = D^{dsc_i\ r_1\ r_2\ a_1}$$
$$1 : (7, 13, 27, 32, 47, 52);$$
$$0 : (0, 10, 20, 24, 25, 30, 40, 44, 45, 50),$$

$$\begin{array}{l} \quad\ 10 \quad 20 \quad\ 40 \\ 4\ \underline{7}\ /\ 17\ /\ 27\ 37\ /\ 47\ 57\ 67\ 77, \\ 1\ \underline{3} \\ \quad\ \underline{6} \\ \quad\ \cdot \end{array}$$

$$D^{dsc_i\ r_1\ r_2\ a_1}_{\underline{000}\ \underline{1}\ 1\ \underline{1}} = r_2.$$

Rezultă

$$A_2 = r_2.$$

Schema de comutație dinamică este reprezentată în figura 11.15. La structurarea ei s-a urmărit folosirea în comun a unor contacte sau grupuri de contacte.

Schema de comutație statică este reprezentată în figura 11.16. Ea prezintă unele particularități.

Astfel releele de timp sînt înlocuite cu module „timp", iar funcția releelor de sens R_d și R_s este preluată de tensiunea culeasă la bornele indusului ($+U$ și $-U$). Potențialele electrice ale periilor indusului înlocuiesc semnalele aferente contactelor r_d și r_s. Diodele D_1 și D_2 din această schemă, conectate cu un capăt

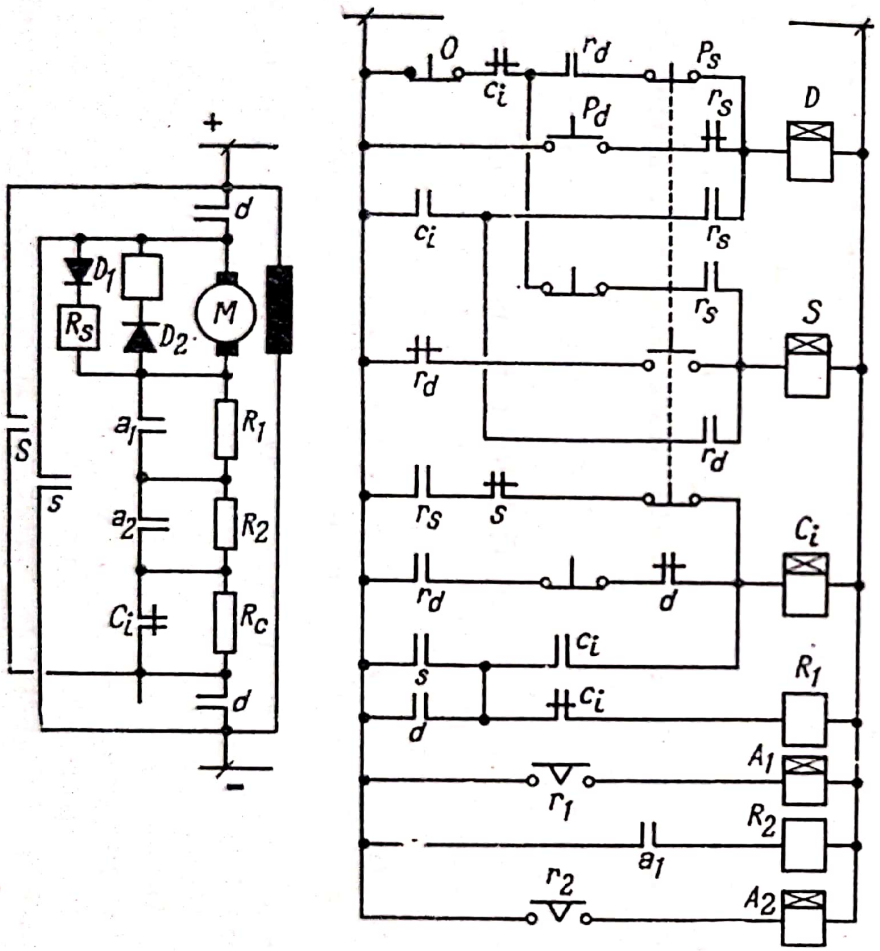


Fig. 11.15. — Schema cu contacte pentru comanda automată, reversibilă, cu frînare prin contraconectare și pornire temporizată a unui motor de c.c.

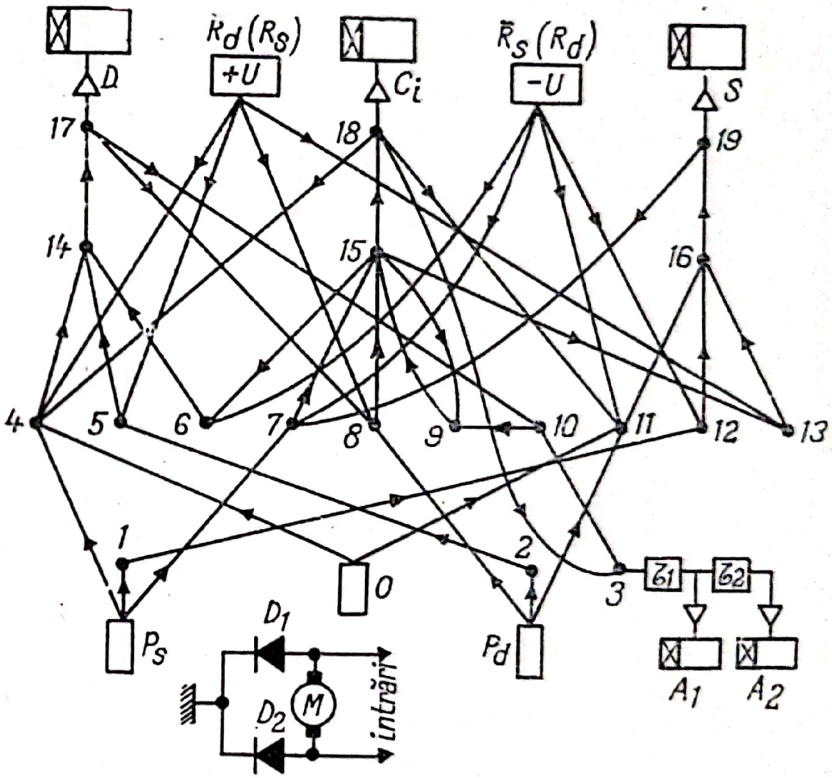


Fig. 11.16. — Schema fără contacte pentru comanda automată reversibilă, cu frînare prin contraconectare și pornire temporizată a unui motor de c.c.

la periile colectoare ale indusului şi cu celălalt capăt la bara emitoarelor tranzistoarelor folosite la realizarea modulelor „nici", fac ca diferenţa de potenţial U să se aplice la intrările tranzistoarelor cu o polaritate sau alta în funcţie de sensul de rotaţie al indusului.

Pragul de acţionare asupra schemei de comutaţie statică depinde, evident, de nivelul tensiunii necesare pentru comanda intrărilor modulelor „nici". Se pot prevedea dispozitive de ajustare a tensiunii U, astfel încît acest prag (tensiune minimă de acţionare) să fie programabil.

În fapt, diodele reprezentînd un scurtcircuit pentru cazul cînd potenţialul este $+U$, pe bazele tranzistoarelor care realizează funcţiile logice „nici" se va aplica numai potenţialul electric $-U$ atunci cînd apare, funcţie de sensul de rotaţie al indusului.

În fine, pentru ridicarea nivelului puterii semnalelor de ieşire se utilizează amplificatoare, care au drept sarcină bobinele de excitaţie ale contaactoarelor D, C_i, S, A_1 şi A_2, sau circuitele de bază ale tiristoarelor ce le înlocuiesc.

În cazul cînd numărul de intrări sau de ieşiri la un modul „nici" depăşeşte pe cel admis prin construcţia modulului, sînt necesare module suplimentare.

Dacă presupunem că modulul „nici" nu admite decît k ieşiri, atunci se folosesc $k-1$ ieşiri normale, iar ieşirea k se aplică unui grup de două module „nici" suplimentare legate în serie. La ieşirea ultimului modul dispunem astfel de încă k ieşiri, astfel încît în total dispunem de $2k-1$ ieşiri.

Similar se poate proceda cînd este depăşit numărul intrărilor pentru modulul „nici".

Remarcăm că, dacă n-am fi folosit reprezentarea schemei de comutaţie statică sub forma de graf orientat, în reprezentarea obişnuită, folosind simbolurile de reprezentare admise în general, ar fi rezultat o schemă foarte complicată, mult mai dificil de modelat experimental pe un simulator logic sau de realizat practic.

Exemplul 11.8 Să se facă sinteza schemei de comandă automată a pornirii unui motor sincron în asincron.

Iniţial, statorul motorului este alimentat la o tensiune redusă prin intermediul unor reactoare.

După un interval de timp programabil, statorul se alimentează la tensiune normală şi în momentul îndeplinirii condiţiilor de sincronizare se conectează excitaţie rotorului. Dacă condiţiile

Scanned with OKEN Scanner

de sincronizare nu sînt îndeplinite în intervalul de timp programat, motorul se oprește și comanda trebuie refăcută.

Comanda de pornire-oprire se dă prin intermediul butoanelor cu revenire P și O, temporizările necesare sînt programate prin reglarea timpului de anclanșare a contactelor cu întîrziere mecanică (relee pendulare), controlate de contactoarele de cuplare la rețeaua de alimentare a statorului, iar controlul sincronizării este făcut printr-un releu conectat în circuitul rotorului.

Butonul de oprire îndeplinește funcția de întrerupător general, ca și contactele releelor de protecție maximală sau de protecție termică ce asigură protecția motorului.

Grupul variabilelor primare va cuprinde variabila P atașată butonului de pornire, contactul releului de sincronizare r_s, variabilele a_t și l_t asociate contactelor temporizate ale contactoarelor A și L, care servesc la cuplarea statorului motorului la rețeaua de alimentare, și variabilele secundare a și l asociate acelorași contactoare. De funcția logică a variabilei atașate butonului O se va ține seama numai la stabilirea formulei finale obținute prin sinteză.

Pentru controlul cuplării excitatricei în circuitul rotorului se impune folosirea unui contactor auxiliar S.

Expresiile logice ale celor trei variabile de ieșire vor fi descrise deci de simbolul de marcare

$$F(A, L, S) = D^{Pa_t\, al_t\, lsr_s},$$

alături, de care se pot considera funcțiile logice ale releului R_s și excitatricei E, care vor depinde de simbolul de marcare

$$F(R_s, E) = D^s$$

Cu aceste date putem trece la stabilirea diagramei de faze, pe care o reprezentăm în figura 11.17.

Stabilirea celor două secvențe de lucru date în această diagramă s-a făcut plecînd de la următoarele considerații. Prin apăsarea butonului de pornire P se excită bobina contactorului A prin care statorul motorului este cuplat la tensiunea redusă de alimentare. Prin aceasta, releul R_s din circuitul rotoric devine excitat prin tensiunea indusă de stator în rotor, întrucît motorul în momentul pornirii se comportă analog unui transformator static.

Dacă se dezacţionează butonul P, starea schemei nu se modifică.

După intervalul de timp programat, prin anclanşarea contactului temporizat al contactorului A se pune sub tensiune contactorul principal L, prin care motorul devine alimentat la tensiunea

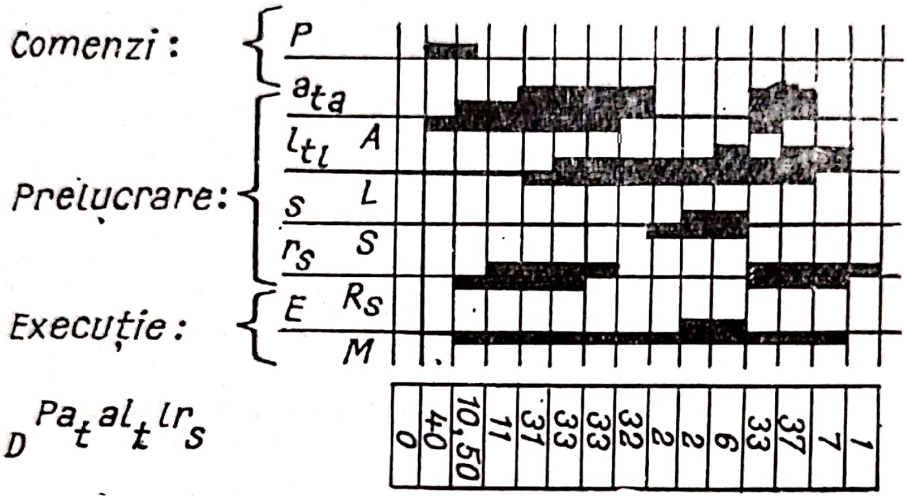


Fig. 11.17. — Diagrama de faze pentru exemplul 11.8.

normală. În această stare, rotorul motorului atinge viteza subsincronă şi curentul din circuitul rotorului are o frecvenţă foarte mică, astfel încît în pauzele dintre două alternanţe releul R_s poate scăpa armătura sa şi deci poate declanşa.

Comportarea contactelor releului de sincronizare este similară contactelor temporizate. Ele declanşează numai atunci cînd frecvenţa tensiunii de excitaţie a acestui releu este suficient de mică (condiţia de sincronism), astfel încît armătura releului să poată scăpa în pauzele dintre pulsurile tensiunii de excitaţie furnizate de rotor.

Dacă în timpul programat releul R_s declanşează, adică sînt îndeplinite condiţiile de sincronizare, schema trece în starea finală. În caz contrar, dacă după scurgerea timpului programat prin temporizarea contactului l_t nu se îndeplinesc condiţiile de sincronism, releul R_s rămîne anclanşat şi drept urmare motorul se opreşte. În acest caz comanda de pornire trebuie repetată.

În urma codificării stărilor date de diagrama de faze din figura 11.17 rezultă

$$A = D^{Pa_t\, al_t\, lr_s}$$
$$1:(10,\ 11,\ 31,\ 33,\ 40,\ 50),\ 0:(0,\ 1,\ 2,\ 6,\ 7,\ 32,\ 37),$$

$$\begin{array}{l} \quad\quad\ 1 \quad\ 20 \\ 40\ \underline{\underline{10}}\ /\ 11\ /\ 30\ 31, \\ 4\ \ \underline{\underline{50}} \\ \quad\ \underline{\underline{14}} \\ \quad\ \cdot\cdot \end{array}$$

$$D^{Pa_t\, al_t\, lr_s}_{\underline{0}\ \underline{0}\ 1\underline{0}\ 0\underline{0}} = a\bar{l},$$

$$\begin{array}{l} \quad\quad\ 10 \quad 20 \\ 40\ \underline{\underline{33}}\ /\ 23\ /\ 13\ 3, \\ \quad\ \underline{\underline{73}} \end{array}$$

$$D^{Pa_t\, a\, l_t\, lr_s}_{\underline{0}\ \underline{1}\ \underline{1}0\,11} = \bar{l}_t\ lr_s,$$

$$\begin{array}{l} \quad\quad\ 1 \quad\ 2 \\ \quad 4\ \underline{\underline{40}}\ /\ 41\ /\ 42\ 43, \\ 10\ \underline{\underline{44}} \\ \quad\ \underline{\underline{50}} \\ 20\ \underline{\underline{54}} \\ \quad\ \underline{\underline{60}} \\ \quad\ \cdot\cdot \end{array}$$

$$D^{Pa_t\, al_t\, lr_s}_{1\ \underline{0}\ \underline{00}\,\underline{00}} = P.$$

Rezultă deci

$$A = (P \cup a\bar{l}_t \cup \bar{l}_t\, lr_s)\ \overline{O}.$$

Pentru L avem

$$L = D^{Pa_t\, al_t\, lr_s}_{1:(2,\ 6,\ 31,\ 32,\ 33,\ 37);\ 0:(0,\ 1,\ 7,\ 10,\ 11,\ 40,\ 50),}$$

$$\begin{array}{l} \quad\quad\ 4 \quad 10 \quad\quad\ 20 \\ 40\ \underline{\underline{2}}\ /\ 6\ /\ 12\ 16\ /\ 22\ 26\ 32\ 36, \\ \quad\ \underline{\underline{42}} \end{array}$$

$$D^{Pa_t\, al_t\, l\, r_s}_{\underline{0}\ \underline{0}\ \underline{00}\,10} = l\bar{r}_s,$$

$$19\ \frac{21}{40}\ \frac{71}{\cdot\cdot}\ \overset{2}{/}\ 33\ \overset{4}{/}\ 35\ 37\ \overset{1}{/}\ 29\ 32\ 34\ 36,$$

$$D^{Pa_t\, al_t\, lr_s}_{\underline{0}\ 1\ \underline{10}\ \underline{01}} = a_t .$$

Rezultă deci

$$L = (l\bar{r}_s \cup a_t)\,\overline{O}\,.$$

Pentru S avem

$$S = D^{Pa_t\, al_t\, lr_s}_{1:(2,\ 6);\ 0:(0,\ 1,\ 7,\ 10,\ 11,\ 31,\ 32,\ 33,\ 37,\ 40,\ 50),}$$

$$40\ \frac{2}{42}\ \overset{4\ \ 10}{/}\ 6\ /\ 12\ 16,$$

$$D^{Pa_t\, al_t\, l\, r_s}_{\underline{0}\ 0\ \underline{00}\ 10} = \bar{a}_t\, l\bar{r}_s .$$

Rezultă deci

$$S = \bar{a}_t\, l\bar{r}_s\, \overline{O}\,.$$

Funcțiile logice ale releului R_s și ale circuitului de excitație a motorului rezultă direct din diagrama de faze

$$R_s = \bar{s} \quad \text{și} \quad E = s.$$

Schemele cu și fără contacte sînt date în figurile 11.18 și 11.19.

O schemă cu contacte îndeplinind funcții similare este dată în [12, p. 285] și de asemenea o astfel de schemă a fost sintetizată, folosind direct formula de interpolare a lui Lagrange, în lucrarea [4, p. 278].

Exemplul 11.9. Să se facă sinteza schemei de comandă automată a unei prese de ambutisat.

Antrenarea poansonului presei se face acționînd o cuplă electromagnetică care cuplează excentricul presei la un arbore în rotație continuă.

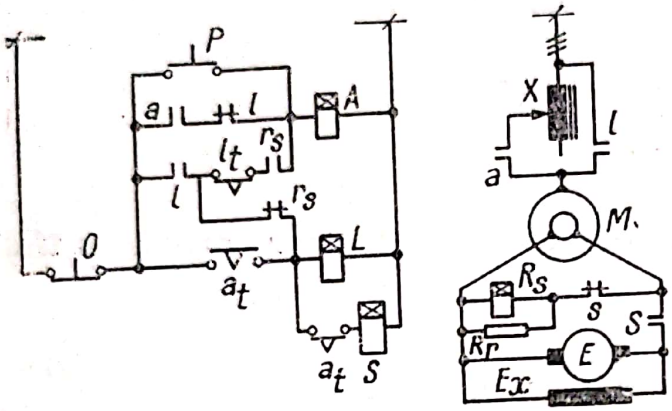


Fig. 11.18. — Comanda automată a pornirii unui motor sincron în asincron.

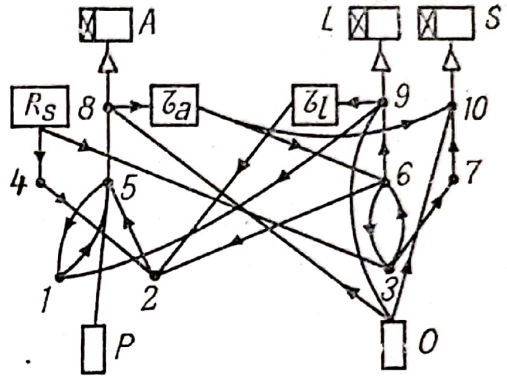


Fig. 11.19. — Comanda automată a pornirii unui motor sincron în asincron.

Specificațiile schemei sînt:

a) Pornirea presei (pentru coborîre) se face numai prin apăsarea simultană pe două butoane B_1 și B_2, pentru a asigura protecția mîinilor operatorului.

b) Eliberarea oricărui buton, B_1 sau B_2 înainte de atingerea unui contact fine de cursă inferior C_1, oprește mișcarea presei.

c) După atingerea contactului C_1 presa continuă mișcarea ridicînd poansonul chiar dacă butoanele B_1 și B_2 se eliberează.

d) Cînd presa ajunge în poziția superioară se atinge contactul C_2 și după aceea presa se oprește.

e) Presa nu trebuie să pornească chiar dacă se mai continuă apăsarea butoanelor B_1 și B_2, deci ea nu execută decît un singur ciclu la o apăsare a butoanelor; un nou ciclu poate începe numai după eliberarea ambelor butoane și darea unei noi comenzi.

f) Atingerea accidentală a contactului C_1 nu trebuie să pornească presa.

Analiza specificațiilor date prin enunțul problemei de sinteză evidențiază următoarele stări distincte, stabile ale schemei

— starea care precede acționării contactului C_1,
— starea care succede acționării contactului C_1,
— starea care succede acționării contactului C_2.

Pentru memorarea acestor stări sînt necesare două variabile secundare.

Considerăm drept variabile secundare variabilele a şi z, asociate variabilelor de ieşire A şi Z. Variabila de ieşire Z o asociem cuplei electromagnetice.

Rezultă deci raportul de stare

$$\frac{B_1B_2C_1 \cdot C_2az}{AZ}$$

Codificat, specificaţiile date prin enunţ, se exprimă astfel:
Specificaţia a)

$$\frac{0}{0} \xrightarrow{B_1=B_2=1} \frac{60}{1} \xrightarrow{z=1} \frac{61}{1}$$

Specificaţia b)

$$61 \xrightarrow{B_1=0} \frac{21}{0}\ ;\quad 61 \xrightarrow{B_2=0} \frac{41}{0}\ ;\quad 61 \xrightarrow{B_1=B_2=0} \frac{1}{0}$$

Specificaţia c)

$$61 \xrightarrow{C_1=1} \frac{71}{3} \xrightarrow{a=1} \frac{73}{3} \xrightarrow{C_1=0} \frac{63}{3}$$

$$73 \xrightarrow{B_1=0} \frac{33}{3}\ ;\quad 73 \xrightarrow{B_2=0} \frac{53}{3}\ ;\quad 73 \xrightarrow{B_1=B_2=0} \frac{13}{3};$$

$$63 \xrightarrow{B_1=0} \frac{23}{3}\ ;\quad 63 \xrightarrow{B_2=0} \frac{43}{3}\ ;\quad 63 \xrightarrow{B_1=B_2=0} \frac{3}{3}.$$

Specificaţia d):

$$3 \xrightarrow{C_2=1} \frac{7}{0} \xrightarrow{a=z=0} \frac{4}{0} \xrightarrow{C_2=0} 0.$$

Scanned with OKEN Scanner

Specificația e)

$$63 \xrightarrow{C_2=1} \frac{67}{2} \xrightarrow{z=0} \frac{66}{2} \xrightarrow{C_2=0} \frac{62}{2} \xrightarrow{B_1=B_2=0} \frac{2}{0} \xrightarrow{a=0} 0.$$

Specificația f):

Examinînd stările ce corespund la $Z = 0$, rezultă:

$$\frac{10}{0}\ ;\ \frac{31}{0}\ ;\ \frac{51}{0}\ ;\ \frac{11}{0}\ ;\ \frac{17}{0}\ ;\ \frac{14}{0}\ ;\ \frac{77}{2}\ ;\ \frac{76}{2}\ ;\ \frac{72}{2}\ ;\ \frac{12}{0}.$$

Pe baza formelor codificate ale stărilor schemei, se deduce următorul simbol de marcare:

$$F(A, Z) = D^{B_1B_2C_1 \cdot C_2\ az}$$

A : 1 : (3, 13, 23, 33, 43, 53, 62, 63, 66, 67, 71, 72, 73, 76, 77)
0 : (0, 1, 2, 4, 7, 10, 11, 12, 14, 17, 21, 31, 41, 51, 60, 61)
Z : 1 : (3, 13, 23, 33, 43, 53, 60, 61, 63, 71, 73)
0 : (0, 1, 2, 4, 7, 10, 11, 12, 14, 17, 21, 31, 41, 51, 62, 66, 67, 72, 76, 77).

Grupele de numere de stare adiacente specifice variabilelor A și Z sînt:

Pentru A

$$\overset{10}{3} \mid \overset{20}{13} \mid 23\ \overset{40}{33} \mid 43\ 53\ 63\ 73 \longrightarrow \underline{000}.011 \rightarrow \overline{C}_2\ az.$$

$$20\ \frac{62}{42}\ \overset{4}{\mid}\ \overset{1}{66} \mid \overset{10}{63\ 67} \mid 72\ 76\ 73\ 77 \longrightarrow \underline{110}.\underline{010} \rightarrow B_1\ a.$$

$$\overset{1}{71} \mid \overset{2}{70} \mid \overset{4}{73\ 72} \mid 75\ 74\ 77\ 76 \longrightarrow 111.\underline{001} \rightarrow B_1B_2C_1.$$

Scanned with OKEN Scanner

Pentru Z

$$\begin{array}{l} \overset{10}{3} \;/\; \overset{20}{13} \;/\; 23\ \overset{40}{33} \;/\; 43\ 53\ 63\ 73 \longrightarrow \underline{000}.011 \rightarrow \overline{C}_2\, az. \\ \overset{1}{60} \;/\; \overset{10}{61} \;/\; 70\ \overset{4}{71} \;/\; 64\ 65\ 74\ 75 \longrightarrow 11\underline{0}.\underline{00}\underline{0} \rightarrow B_1B_2\overline{a}. \end{array}$$

Rezultă expresiile logice :

$$A = \overline{C}_2 az \cup B_1(a \cup B_2C_1)$$

$$Z = \overline{C}_2 az \cup B_1B_2\overline{a}.$$

Ele corespund schemei date în figura 11.20 *a* şi 11.20 *b* şi cu rezultatele date în lucrarea [33. p. 312]

Contactele B_1 şi B_2 pot fi folosite în comun de funcţiile lui A şi Z întrucît circuitele logice suplimentare apărute au valoare logică 0.

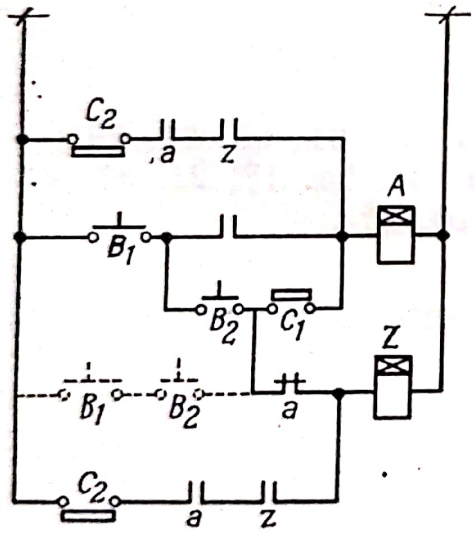


Fig. 11.20a. — Schema de comutaţie dinamică pentru exemplul 11.9.

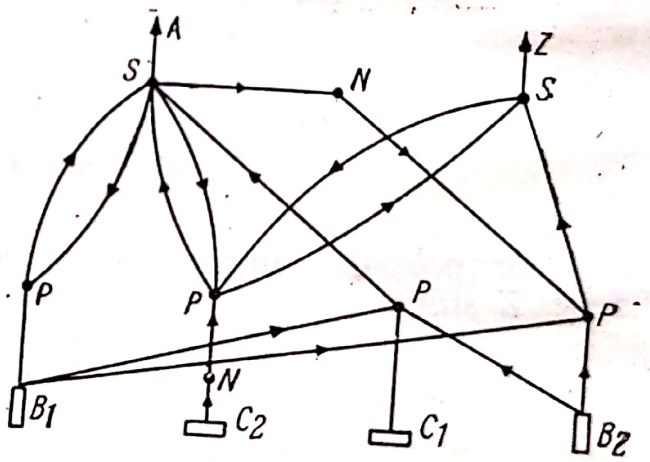


Fig. 11.20b. — Schema de comutaţie statică pentru exemplul 11.9.

Într-adevăr pentru A, avem

$$C_1\overline{a}az\overline{C}_2 = 0$$

iar pentru Z

$$\overline{a}C_1aB_1 = 0 \quad \text{şi} \quad \overline{a}az\overline{C}_2 = 0$$

Niciunul dintre contactele $\overline{C}_2$, a și z nu pot fi folosite în comun de funcțiile logice A și Z întrucît

— valoarea logică a circuitelor nou create este diferită de zero,

— noile circuite, determină apariția unor noi numere de stare. dintre care cel puțin unul este interzis funcțiilor logice A sau Z.

11.2. SINTEZA SCHEMELOR DE COMUTAȚIE PENTRU ACȚIONĂRI ELECTRICE CU CINEMATICĂ PROGRAMATĂ

În cele ce urmează se vor trata o serie de exemple de acționări electrice cu grupuri de mașini a căror funcționare este intercondiționată, acționări electrice cu motoare funcționînd după diagrame de lucru programabile sau acționări electrice cu motoare a căror funcționare este controlată prin dispozitive de control, cum ar fi limitatoarele de cursă. În toate cazurile ne referim numai la sistemele secvențiale asincrone.

Exemplul 11.10. Să se facă sinteza schemei de comandă automată a unei acționări electrice cu trei electromotoare blocate în cascadă (transportor cu bandă din trei secțiuni).

Comenzile de pornire-oprire sînt date printr-un singur buton de pornire P și unul de oprire O, ambele cu revenire.

În cazul blocării în cascadă, conectarea motoarelor la rețeaua electrică de alimentare se realizează cu ajutorul a trei contactoare : A, B și C. Ordinea de pornire este C, B, A, iar cea de oprire este inversă, adică A, B, C.

Acționarea butonului de oprire O trebuie să aibă loc pe toată durata opririi electromotoarelor și ea este prioritară în raport cu comanda dată prin butonul de pornire P.

Cu aceste precizări, sîntem în măsură să trasăm diagrama de faze din figura 11.21.

Folosind stările codificate pe această diagramă, se obțin următoarele rezultate

— pentru A :

$$A = D^{OPabc}_{1:(3,7);0:(0,1,10,20,21,23,27,30,31,33,37),}$$

4 1 10
3 / 7 / 2 6 / 13 17 12 16,

$$A = D^{OPabc}_{0\underline{00}1\underline{1}} = \overline{O}b;$$

— pentru B :

$$B = D^{OPabc}_{1:(1,3,7,27,37);0:(0,10,20,21,23,30,31,33),}$$

2 4 10
1 / 3 / 5 7 / 11 13 15 17,

$$D^{OPabc}_{0\underline{000}1} = \overline{O}c,$$

10 1 2
20 $\frac{27}{7}$ / 37 / 26 36 / 25 35 24 34,

$$D^{OPabc}_{\underline{1}0\underline{11}} = a,$$

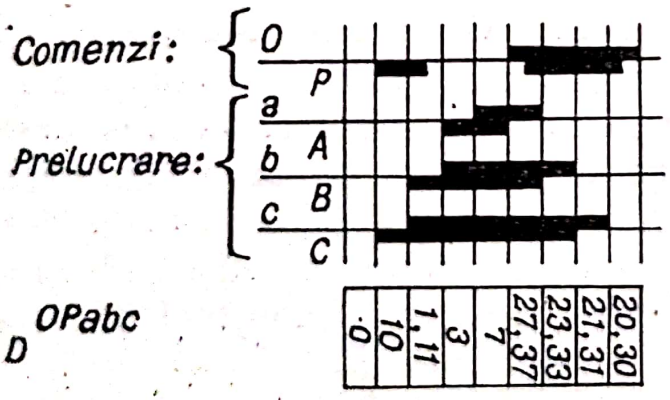


Fig. 11.21. — Diagrama de faze pentru exemplul 11.10.

rezultă

$$B = \overline{O}c \cup a;$$

— pentru C

$$C = D^{OPabc}_{1:(1,3,7,10,23,27,33,37);0:(0,20,21,30,31)},$$

$$\begin{matrix} 2 & 4 & 10 \\ 1 / 3 / 5\ 7 / 11\ 13\ 15\ 17, \end{matrix}$$

$$D^{OPabc}_{\underline{00}0\underline{0}1} = \overline{O}c,$$

$$\begin{matrix} 1 & 2 & 4 \\ 10 / 11 / 12\ 13 / 14\ 15\ 16\ 17, \end{matrix}$$

$$D^{OPabc}_{01\underline{000}} = \overline{O}P,$$

$$\begin{matrix} & 10 & 4 & 1 \\ 20 & \underline{23} / 33 / 27\ 37 / 22\ 32\ 26\ 36, \\ & 3 \end{matrix}$$

$$D^{OPabc}_{\underline{10}0\underline{1}1} = b;$$

rezultă deci

$$C = \overline{O}(P \cup c) \cup b.$$

Schemele cu și fără contacte care corespund formulelor logice deduse sînt date în figura 11.22.

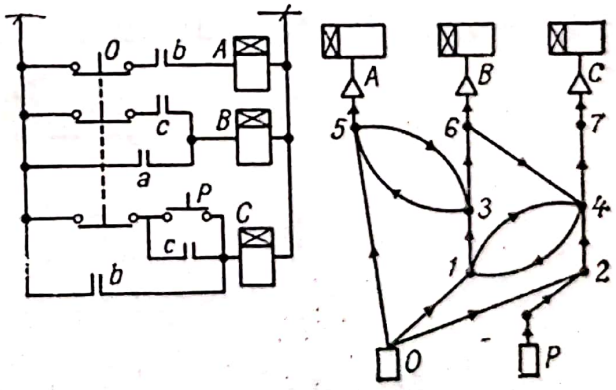


Fig. 11.22. — Blocarea în cascadă a trei electromotoare (transportor cu bandă).

Această problemă mai este rezolvată (sinteza numai a schemei cu contacte) în lucrările [4, p. 235] și [5].

Exemplul 11.11. O instalație tehnologică este dotată cu trei electromotoare, acționate prin trei contactoare: Z_1, Z_2 și Z_3.

Electromotoarele se cuplează direct la rețeaua de alimentare și au un singur sens de rotație. Pentru pornirea și oprirea fiecărui electromotor este prevăzut cîte un întrerupător: I_1, I_2 și I_3.

Pornirea și oprirea electromotoarelor se face într-o anume ordine, aceeași pentru pornire și pentru oprire. Astfel, motorul 2 pornește după τ_1 s în urma motorului 1 și al treilea după τ_2 s în urma motorului 2.

Comanda de pornire-oprire, dată manual, se memorează și se execută atunci cînd sînt îndeplinite condițiile specificate mai sus.

Întrucît starea de funcționare a motorului 1 o condiționează pe cea a motorului 2 în același fel în care starea motorului 2 o condiționează pe cea a motorului 3, se poate face sinteza numai pentru schema de comutație a primelor două motoare. Prin permutare se vor putea apoi deduce funcțiile logice pentru cel de-al treilea motor.

Considerăm drept variabile primare variabilele asociate celor trei întrerupătoare de comandă manuală. Variabilele asociate contactelor temporizate controlate de contactoarele Z_1 și Z_2 le notăm prin zt_1 și zt_2. Variabilele secundare și cele de ieșire le asociem contactoarelor care cuplează la rețeaua de alimentare cele trei electromotoare.

Rezultă următorul simbol de marcare:

$$F(Z_1, Z_2) = D^{I_1 I_2 z_{t_1} z_1 z_{t_2} z_2}$$

și diagrama de faze din figura 11.23.

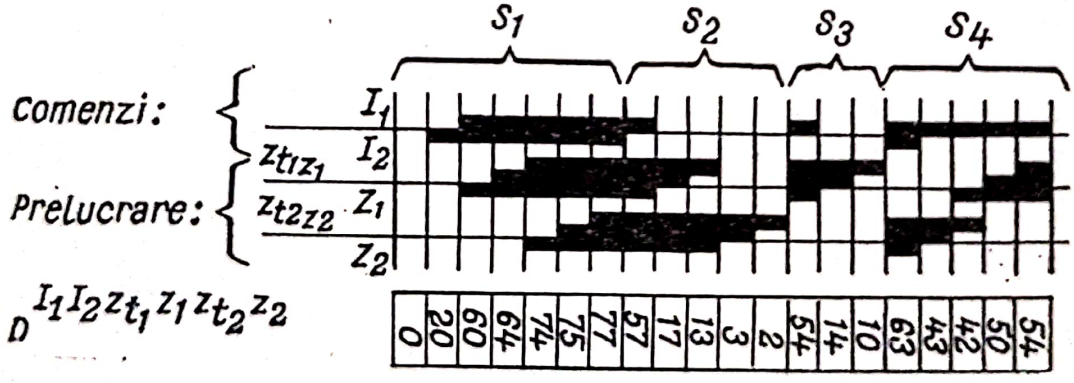


Fig. 11.23. — Diagrama de faze pentru exemplul 11.11.

Secvența S_1 se referă la starea $Z_1 = Z_2 = 0$, cînd se dă comanda $I_2 = 1$ și apoi $I_1 = 1$.

Secvența S_2 se referă la starea $Z_1=Z_2=1$, cînd se dă comanda $I_2 = 0$, apoi $I_1 = 0$; secvența S_3 se referă la starea $Z_1=1$, $Z_2= 0$, cînd se dă comanda $I_1 = 0$ iar secvența S_4 se referă la starea $Z_1 = 0$, $Z_2 = I_1 = I_2 = 1$, cînd se dă comanda $I_2 = 0$.

Rezultă următoarele funcții logice:

— pentru Z_1:

$$Z_1 = D^{I_1 I_2 z_{t_1} z_1 z_{t_2} z_2}_{1:(42,50,54,57,60,64,74,75,77),\ 0:(0,2,3,10,13,14,17,20,43,63),}$$

$$20\ \frac{42}{62}\ \overset{2}{/}\ 40\ \overset{4}{/}\ 46\ 44\ \overset{10}{/}\ 52\ 50\ 56\ 54,$$

$$D^{I_1 I_2 z_{t_1} z_1 z_{t_2} z_2}_{1\ \underline{0}\ \underline{0}\ \underline{0}\ \underline{1}} = I_1 \bar{z}_2,$$

$$10\ \frac{57}{47}\ \overset{20}{/}\ 77\ \overset{2}{/}\ 55\ 75\ \overset{1}{/}\ 56\ 76\ 54\ 74,$$

$$D^{I_1 I_2 z_{t_1} z_1 z_{t_2} z_2}_{1\ \underline{0}\ \underline{1}\ 1\ \underline{1}\ \underline{1}} = I_1 z_1;$$

rezultă deci

$$Z_1 = I_1(z_1 \cup \bar{z}_2);$$

— pentru Z_2

$$Z_2 = D^{I_1 I_2 z_{t_1} z_1 z_{t_2} z_2}_{1:(13,17,57,63,74,75,77),\ 0:(0,2,3,10,14,20,42,43,50,54,60,64),}$$

$$20\ \frac{13}{33}\ \overset{4}{/}\ 17\ \overset{40}{/}\ 53\ 57\ \overset{2}{/}\ 11\ 15\ 52\ 55,$$

$$D^{I_1 I_2 z_{t_1} z_1 z_{t_2} z_2}_{\underline{0}\ \underline{0}\ 1\ \underline{0}\ \underline{1}\ 1} = z_{t_1} z_2,$$

$$40\ \frac{63}{23}\ \overset{2}{/}\ 65\ \overset{10}{/}\ 73\ 75\ \overset{4}{/}\ 67\ 61\ 77\ 71,$$

$$D^{I_1 I_2 z_{t_1} z_1 z_{t_2} z_2}_{\underline{1}\ 1\ \underline{0}\ \underline{0}\ \underline{1}\ 1} = I_2 z_2,$$

$$40\ \frac{74}{34}\ \overset{2}{/}\ 76\ \overset{1}{/}\ 75\ 77\ \overset{4}{/}\ 70\ 72\ 71\ 73,$$

$$D^{I_1 I_2 z_{t_1} z_1 z_{t_2} z_2}_{\underline{1}\,1\,1\,\underline{1}\,\underline{0}\,\underline{0}} = I_2 z_{t_1};$$

rezultă deci

$$Z_2 = I_2(z_2 \cup z_{t_1}) \cup z_{t_1} z_2.$$

Prin permutare se obţine

$$Z_3 = I_3(z_3 \cup z_{t_2}) \cup z_{t_2} z_3.$$

Pe baza rezultatelor obţinute s-a structurat schema de comutaţie dinamică din figura 11.24.

Aplicarea metodei matriceale pentru sinteza acestei scheme se poate găsi în [23, p. 37—38].

Efectuînd sinteza prin grupe de O, se ajunge la

$$Z_1 = \overline{\bar{I}_1 \cup z_2 \bar{z}_{t_1}},$$

$$Z_2 = \overline{\bar{I}_2 \bar{z}_2 \cup \bar{I}_2 \bar{z}_{t_1} \cup \bar{z}_{t_1} \bar{z}_2},$$

$$Z_3 = \overline{I_3 \bar{z}_3 \cup \bar{I}_3 \bar{z}_{t_2} \cup \bar{z}_{t_2} \bar{z}_3}.$$

Această soluţie este complementul celei obţinute prin sinteza prin grupe de 1, dar ea e mai convenabilă pentru structurarea schemei modulare UNILOG. Schema este dată în figura 25 şi corespunde ecuaţiilor

$$Z_1 = \bar{I}_1 \top (\bar{z}_2 \top z_{t_1}),$$

$$Z_2 = (I_2 \top z_2) \top (I_2 \top z_{t_1}) \top (z_{t_1} \top z_2).$$

$$Z_3 = (I_3 \top z_3) \top (I_3 \top z_{t_2}) \top (z_{t_2} \top z_3).$$

Scanned with OKEN Scanner

Exemplul 11.12. Să se facă sinteza schemei de comandă automată a dispozitivului de avans longitudinal folosit la un strung revolver.

La acționarea butonului de comandă P, cuplajul electromagnetic comandat devine acționat, această stare fiind memorată după dezacționarea acestui buton.

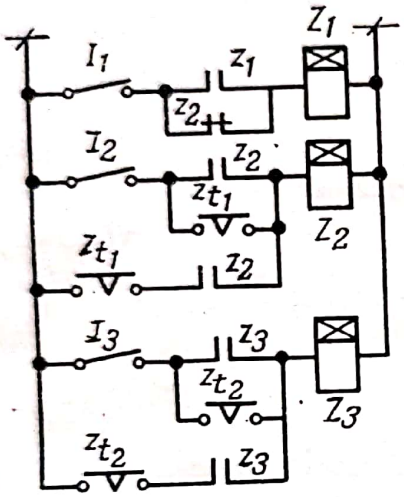


Fig. 11.24. — Schema cu contacte pentru comanda automată a unei instalații tehnologice cu trei electromotoare blocate în cascadă.

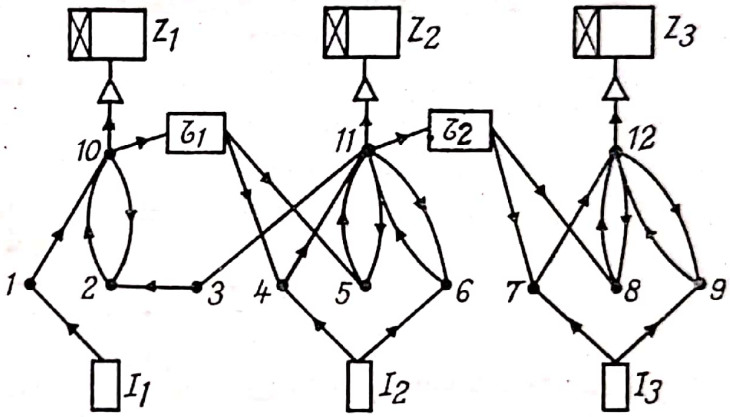


Fig. 11.25. — Schema modulară UNILOG pentru comanda automată a unui grup de trei electromotoare.

La o nouă acționare a butonului P, cuplajul electromagnetic comandat devine neacționat, această stare fiind memorată după dezacționarea acestui buton.

Cele patru stări stabile ale schemei care rezultă din enunțul problemei de sinteză pot fi realizate cu ajutorul contactorului cuplei electromagnetice L și a unui contactor auxiliar A. Cu variabilele secundare asociate contactelor acestor contactoare se pot forma $2^2=4$ combinații, care pot fi folosie pentru realizarea stărilor stabile ale schemei.

Specificațiile date prin enunț și precizările făcute mai sus ne conduc la stabilirea diagramei de faze din figura 11.26. Numerele de stare care rezultă din codificarea diagramei de faze ne conduc la stabilirea următoarei soluții a problemei :

Scanned with OKEN Scanner

— pentru A :

$$A = D^{Pal}_{(1,3,6,7)} \qquad \overset{2}{1/3} \to D^{Pal}_{00\underline{1}} = \overline{P}l,$$

$$\overset{1}{6/7} \to D^{Pal}_{11\underline{0}} = Pa,$$

$$A = Pa \cup \overline{P}l;$$

— pentru L :

$$L = D^{Pal}_{(1,3,4,5)} \qquad \overset{2}{1/3} \to D^{Pal}_{00\underline{1}} = \overline{P}l,$$

$$\overset{1}{4/5} \to D^{Pal}_{10\underline{0}} = P\overline{a},$$

$$L = P\overline{a} \cup \overline{P}l.$$

Sinteza acestei scheme cu contacte prin metoda folosirii directe a formulei de interpolare a lui Lagrange face obiectul lucrării [6, p. 64—66].

Schema citată, la fel ca schema cu contacte dată în figura 11.27*a*, numai cu linii pline, prezintă hazard de continuitate.

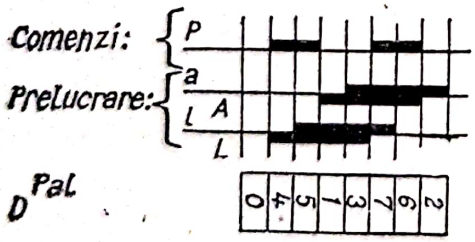


Fig. 11.26. — Diagrama de faze pentru exemplul 11.12.

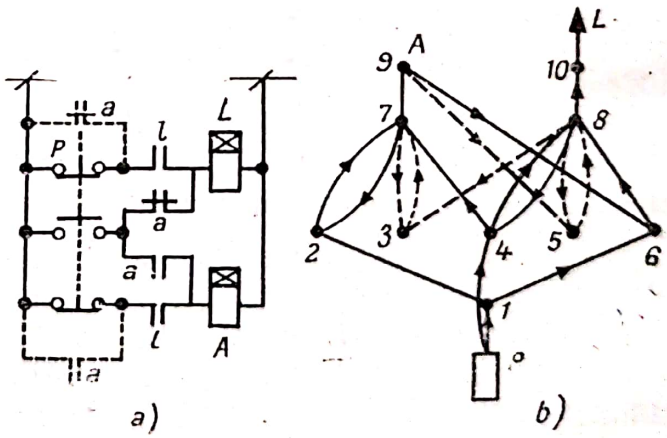


Fig. 11.27. — Comanda automată a avansului longitudinal pentru un strung revolver.

Astfel, în situaţia cînd toate contactele P sînt desfăcute pe durata foarte scurtă a comutării lor, circuitele de comandă se

întrerup, astfel încît există posibilitatea să nu fie satisfăcute specificațiile date prin enunț.

Eliminarea acestui hazard se poate face fie reglînd adecvat timpii de anclanșare și declanșare ai contactoarelor, fie adăugînd termeni redundanți în expresiile logice care descriu circuitele acestor contactoare.

Prezența hazardului de continuitate este evidențiată de grupele de adiacență formate. Astfel, dacă aceste grupe nu au nici un număr de stare comun, așa cum este cazul nostru, schema prezintă hazard de continuitate. Pentru eliminarea acestui hazard se impune adăugarea unei grupe de numere de stare adiacente redundante, grupă care să conțină cel puțin cîte un număr de stare comun grupelor de adiacență care determină implicanții primi ai expresiilor logice.

În exemplul nostru, grupele redundante, necesare pentru eliminarea hazardului de continuitate, sînt:

— pentru A: $\overset{4}{3\,/\,7} \rightarrow D^{Pal}_{\underline{0}11} = al,$

— pentru L: $\overset{4}{1\,/\,5} \rightarrow D^{Pal}_{\underline{0}01} = \bar{a}l.$

Expresiile logice care corespund schemei completate cu linii punctate în figura 9.26, schemă care nu prezintă hazard de continuitate, vor fi

$$A = Pa \cup \bar{P}l \cup (al),$$

$$L = P\bar{a} \cup \bar{P}l \cup (\bar{a}l).$$

Termenii redundanți sînt scriși între paranteze, iar în schemele din figura 11.27. sînt desenați cu linii întrerupte.

Schema modulară UNILOG s-a structurat folosind formulele

$$A = \overline{(\bar{P} \top \bar{a}) \top (P \top \bar{l}) \top (\bar{a} \top \bar{l})},$$

$$L = \overline{(\bar{P} \top a) \top (P \top \bar{l}) \top (a \top \bar{l})}.$$

Facem o primă observaţie că prin transformări se poate reduce numărul de contacte necesare îndeplinirii funcţiilor schemei, ajungîndu-se final la o schemă similară celei date în figura 9.2*c* care conţine numai cinci contacte.

Mai facem observaţia că funcţia butonului de comandă cu contacte multiple poate fi preluată de contactele unui contactor auxiliar, comandat printr-un contact unic al butonului *P*.

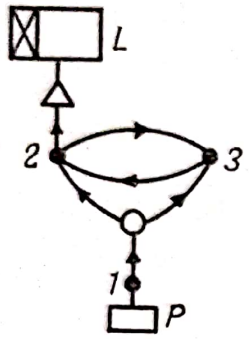


Fig. 11.28. — Schemă suplimentară pentru exemplul 11.12.

De asemenea funcţia logică îndeplinită de această schemă se poate realiza cu un singur circuit basculant bistabil cu intrare unică (prin condensatoare şi diode), ca în figura 11.28.

Exemplul 11.13. Să se facă sinteza unei scheme de comandă automată a acţionării unei mese oscilante pentru o freză sau un *scheping*.

Masa cu mişcare oscilantă este acţionată de un electromotor cuplat la reţeaua de alimentare prin contactoarele *D* şi *S*, cîte unul pentru fiecare sens de rotaţie.

Controlul mişcării oscilante este asigurat prin trei circuite basculante bistabile (memorii), comandate prin microîntrerupătoare. Unul dintre aceste circuite, căruia i se asociază variabila de ieşire *P*, controlează prin cele două intrări ale lui, *M* şi *O*, starea de mers şi cea de repaus a mesei oscilante.

Al doilea circuit basculant bistabil, căruia i se asociază variabila de ieşire *L*, controlează prin cele două intrări ale lui L_a şi L_s (limitatoare de cursă) sensul de rotaţie a mesei oscilante.

Al treilea circuit basculant bistabil, căruia i se asociază variabila de ieşire *R*, controlează prin cele două intrări ale sale, şi anume L_0 limitator de cursă pentru oprirea de lucru şi *A* buton pentru oprirea de avarie sau urgenţă, starea de repaus a mesei oscilante, adică repausul de lucru şi cel de avarie sau de urgenţă.

În figura 11.29 este dată amplasarea tuturor acestor contacte, precum şi a camei *C*, care le acţionează.

Specificaţiile schemei sînt :

1. Dacă masa oscilantă se găseşte în stare de repaus de lucru ($R = 1$), ceea ce se întîmplă atunci cînd cama *C* (fig. 11.29) apasă limitatorul *A*, apăsarea pe butonul de mers *M* (mers) care corespunde stării binare $P = 1$, determină punerea în mişcare a

mesei oscilante în sensul de rotație spre dreapta ($D = 1$). Mișcarea continuă pînă ce cama C atinge limitatorul L_s ($L = 1$) și apoi sensul de rotație se inversează. Mișcarea spre stînga continuă apoi pînă ce cama C atinge limitatorul L_d ($L = 0$), cînd are loc din nou inversarea sensului de rotație etc.

2. Oprirea de lucru a mesei oscilante se comandă prin apăsarea butonului O ($P=0$). Prin aceasta masa oscilantă continuă să execute mișcarea oscilantă pînă în momentul cînd este atins de către cama C limitatorul L_d ($L = 0$), cînd mișcarea se continuă spre stînga, iar la atingerea lui A ($R = 1$) motorul de acționare este oprit.

3. Oprirea de urgență sau cea de avarie trebuie să aibă loc în momentul cînd se apasă butonul A ($R = 1$),indiferent de poziția în care se găsește masa oscilantă.

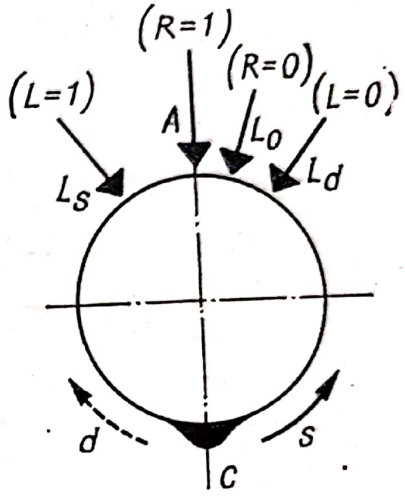


Fig. 11.29. — Masa oscilantă a unei freze.

Circuitul basculant bistabil pentru oprirea de avarie poate fi astfel dimensionat, încît la dispariția și apoi la reapariția tensiunii de alimentare el să capete starea : $R = 1$, ceea ce asigură și protecția de tensiune nulă a schemei.

4. Repunerea în funcțiune după o oprire de urgență sau de avarie se face prin apăsarea lui P și a butonului P_0 ($R = 0$), buton care șuntează contactele limitatorului L_0. În acest fel masa oscilantă intră din primul moment în mersul normal.

Sensul de rotație dorit după oprirea de urgență sau de avarie se poate comanda prin apăsarea butoanelor cu revenire ce dublează comanda limitatoarelor de sens L_d și L_s, adică a butoanelor P_d și P_s.

Dacă nu se face nici o comandă specială și oprirea de urgență a fost comandată prin butonul de avarie A, la punerea în funcțiune a frezei sensul de rotație va fi cel anterior opririi de urgență sau avarie.

Ținînd seama de specificațiile problemei de sinteză întocmim diagrama de faze din figura 11.30. Pentru a comenta stabilirea acestei diagrame ne vom referi la raportul de stare

$$\frac{PLR(ds)}{DS} = \frac{n}{DS}.$$

Succesiunea stărilor secvenţei de mers normal a frezei, care satisface specificaţia 1), este

$$\frac{24}{10} \to \frac{26}{10} \to \frac{22}{10} \to \frac{32}{00} \to \frac{30}{01} \to \frac{31}{01} \to \frac{21}{00} \to \frac{20}{10} \to (22) \cdots$$

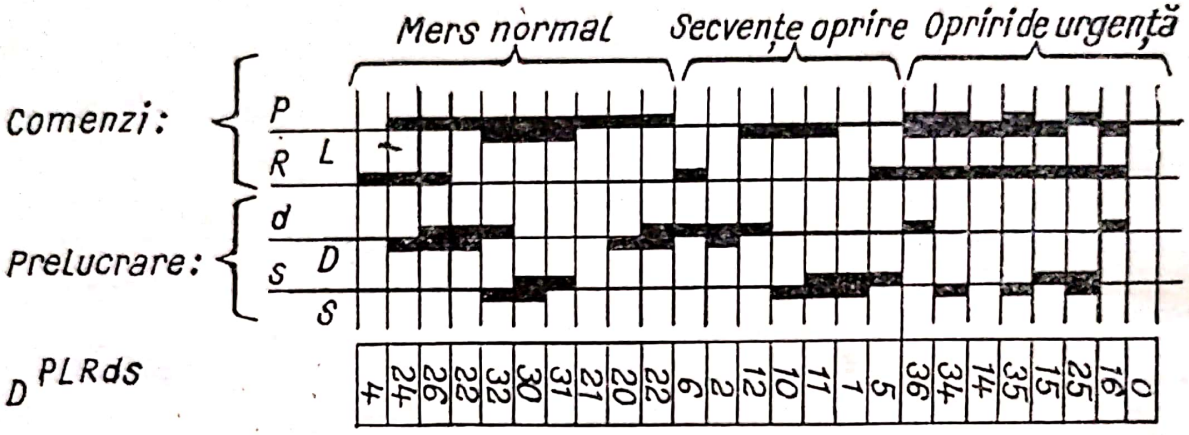


Fig. 11.30. — Diagrama de faze pentru exemplul 11.13.

Apăsarea butonului O, potrivit specificaţiei 2), conduce la desfăşurarea următoarelor secvenţe de oprire funcţie de momentul cînd se dă această comandă:

$$(24) \to (4); \quad (26) \to \frac{6}{00} \to \frac{4}{00}; \quad (22) \to \frac{2}{10} \to \frac{12}{00} \to \frac{10}{01} \to \frac{11}{01} \to \frac{1}{01} \to$$

$$\to \frac{5}{00} \to (4); \quad (32) \to (12); \quad (30) \to (10); \quad (31) \to (11); \quad (21) \to (1);$$

$$(20) \to \frac{0}{00} \to (1);$$

Oprirea de urgenţă, potrivit specificaţiei 3), are loc prin apăsarea lui A ($R = 1$) în oricare dintre stările de lucru ale schemei. O dată cu apăsarea lui A, printr-o legătură mecanică sau electrică (semnal aplicat la intrarea de ştergere a circuitului basculant P) este făcut şi $P = O$. Această legătură nu este şi inversă, adică apăsarea lui O nu influenţează circuitul basculant R.

Stările de blocare determinate de această acționare sînt:

$$(24) \to (4);\ (26) \to (6);\ (22) \to (26) \to (6);$$

$$(32) \nearrow (22) \atop \searrow \underline{36} \nearrow (26) \atop 00 \quad \searrow \underline{34} \to \underline{14} \atop 01 \quad 00;$$

$$(30) \nearrow (10) \to (14) \atop \searrow (34)$$

$$(21) \nearrow (1) \atop \searrow \underline{25} \atop 01 \to (5);$$

$$(31) \nearrow (11) \to (1) \atop \searrow \underline{35} \to \underline{15} \to (14) \atop 01 \quad 00$$

$$(6) \to (4);\quad (20) \nearrow (24) \to (4) \atop \searrow (0)$$

$$(5) \to (4);\ (12) \to \underline{16} \atop 00 \to (14);\ (10) \to (14).$$

Rezultă deci că în urma opririi de urgență schema se poate găsi în stările (4) sau (14).

Repunerea în funcțiune a frezei după oprirea de urgență conform specificației 4), prin apăsarea butonului de pornire P, conduce la secvențele

$$(4) \to (24);\quad (14) \to (34) \to (35) \to (25) \to (21).$$

Sensul de rotație dorit se alege basculînd în mod corespunzător circuitul L cu ajutorul butoanelor P_d și P_s, care dublează comenzile limitatoarelor L_d și L_s. Astfel, dacă $L = 0$, starea dinaintea apăsării lui P este (4) și conduce prin intermediul stării (24) la rotația spre dreapta, iar dacă $L = 1$, starea inițială este (14) și prin apăsarea lui P devine (34), ceea ce determină rotirea mesei oscilante spre stînga.

Dacă butonul de dublare a comenzii lui L_0, adică P_0, nu a fost acționat, atunci, în cazul cînd comanda de pornire se dă din starea (14), are loc o mișcare spre stînga a mesei oscilante pînă ce cama C atinge limitatorul L_0 și abia apoi freza intră în mersul normal.

În diagrama de faze din figura 11.30, s-a redat secvenţa de mers normal a frezei şi numai stările secvenţelor de oprire care diferă de stările secvenţei normale. S-a urmărit ca această diagramă să evidenţieze toate stările posibile ale schemei cel puţin o dată.

Codificarea diagramei de faze ne conduce la următoarele simboluri de marcare:

$$F(D) = D^{PLRds}_{\substack{1:(2,\,20,\,22,\,24,\,26),\\ 0:(0,\,1,\,4,\,5,\,6,\,10,\,11,\,12,\,14,\,15,\,16,\,21,\,25,\,30,\,31,\\ \quad 32,\,34,\,35,\,36),}}$$

$$F(S) = D^{PLRds}_{\substack{1:(1,\,10,\,11,\,25,\,30,\,31,\,34,\,35)\\ 0:(0,2,\,4,5,\,6,\,12,\,14,\,15,\,16,\,20,\,21,\,22,\,24,\,26,32,\,36).}}$$

Expresia booleeană minimală pentru $F(D)$ rezultă din

$$\begin{array}{l} \overset{20}{2} \,/\, \overset{1}{22} \,/\, 3\,23 \end{array} \qquad D^{PLRds}_{\underline{0}\,0\,0\,1\,\underline{0}} = \bar{L}\bar{R}d,$$

$$\begin{array}{l} \overset{4}{20} \,/\, \overset{2}{24} \,/\, 22\,26, \end{array} \qquad D^{PLRds}_{1\,0\,\underline{0}\,\underline{0}\,0} = P\bar{L}\bar{s},$$

rezultă deci

$$F(D) = \bar{L}(P\bar{s} \cup \bar{R}d) = \bar{l}(p\bar{s} \cup \bar{r}d).$$

Similar, pentru $F(S)$ obţinem

$$\begin{array}{l} \overset{10}{1} \,/\, \overset{2}{11} \,/\, 3\,13 \end{array} \qquad D^{PLRds}_{0\,\underline{0}\,0\,\underline{0}1} = \bar{P}\bar{R}s,$$

$$\begin{array}{l} \overset{20}{10} \,/\, \overset{1}{30} \,/\, 11\,31 \end{array} \qquad D^{PLRds}_{\underline{0}\,1\,0\,0\underline{0}} = L\bar{R}\bar{d},$$

$$\begin{array}{l} \overset{10}{25} \,/\, \overset{2}{35} \,/\, 27\,37 \end{array} \qquad D^{PLRds}_{1\,\underline{0}\,1\,\underline{0}1} = PRs,$$

$$\begin{array}{l} \overset{1}{34} \,/\, \overset{4}{35} \,/\, 30\,31 \end{array} \qquad D^{PLRds}_{1\,1\,\underline{1}\,0\underline{0}} = PL\bar{d};$$

rezultă deci

$$F(S) = P(Rs \cup L\bar{d}) \cup \bar{R}(L\bar{d} \cup \bar{P}s) = p(rs \cup l\bar{d}) \cup (l\bar{d} \cup \bar{p}s)\bar{r}.$$

Avînd în vedere că variabilele P, L și R reprezintă ieșirile celor trei memorii folosite și în același timp ele reprezintă și variabile primare pentru circuitele contactorului D și S, expresiile logice pentru $F(D)$ și $F(S)$ au fost transcrise folosind literele mici p, l și r, ele reprezentînd variabilele de intrare (primare) pentru aceste circuite.

Structurarea schemelor cu și fără contacte după expresiile logice minimale obținute nu prezintă nici o particularitate. Ele sînt date în figurile 11.31 și 11.32.

Pentru comparație, se poate consulta rezolvarea acestei probleme prin metoda matriceală obișnuită, dată în [25, p. 151—159]

Exemplul 11.14. Să se facă sinteza schemei de comandă automată a (acționării electrice a mecanismului portcuțit a unei raboteze.

Acționarea folosește un electromotor pentru deplasarea traversei portcuțit și unul pentru blocarea acesteia.

Comanda reversibilă a acestor electromotoare este asigurată prin patru contactoare: R, C, D și B, respectiv pentru ridicarea, coborîrea, deblocarea și blocarea traversei portcuțit a rabotezei.

Controlul comenzilor este asigurat prin butoanele cu revenire P_r și P_c pentru comenzile de pornire-oprire a electromotorului de deplasare a traversei și printr-un circuit cu memorie, comandat la intrare de două limitatoare pentru blocarea traversei în pozițiile limită.

Traversa portcuțit este inițial blocată într-o anumită poziție ($L = 0$). Pentru deplasarea ei într-un sens sau altul trebuie să se realizeze mai întîi deblocarea ($L = 1$) și apoi deplasarea în sensul comandat. Ajunsă în poziția dorită, traversa portcuțit trebuie blocată din nou.

Stările secvențelor de ridicare, respectiv de coborîre a traversei portcuțit, sînt date de diagrama de faze din figura 11.33. Adiacența stărilor acestor secvențe asigură continuitatea trecerilor impuse de lucrul schemei.

Cunoscînd stările obligate ale circuitului de comutație din diagrama de faze codificată, stabilim următoarele simboluri de mar-

Scanned with OKEN Scanner

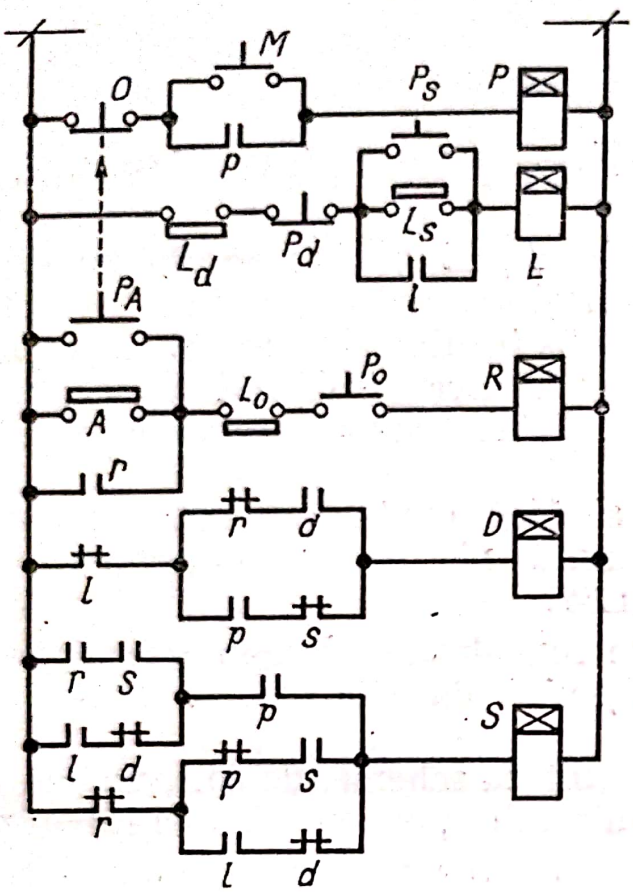


Fig. 11.31. — Schema cu contacte pentru comanda automată a acţionării electrice a unei freze.

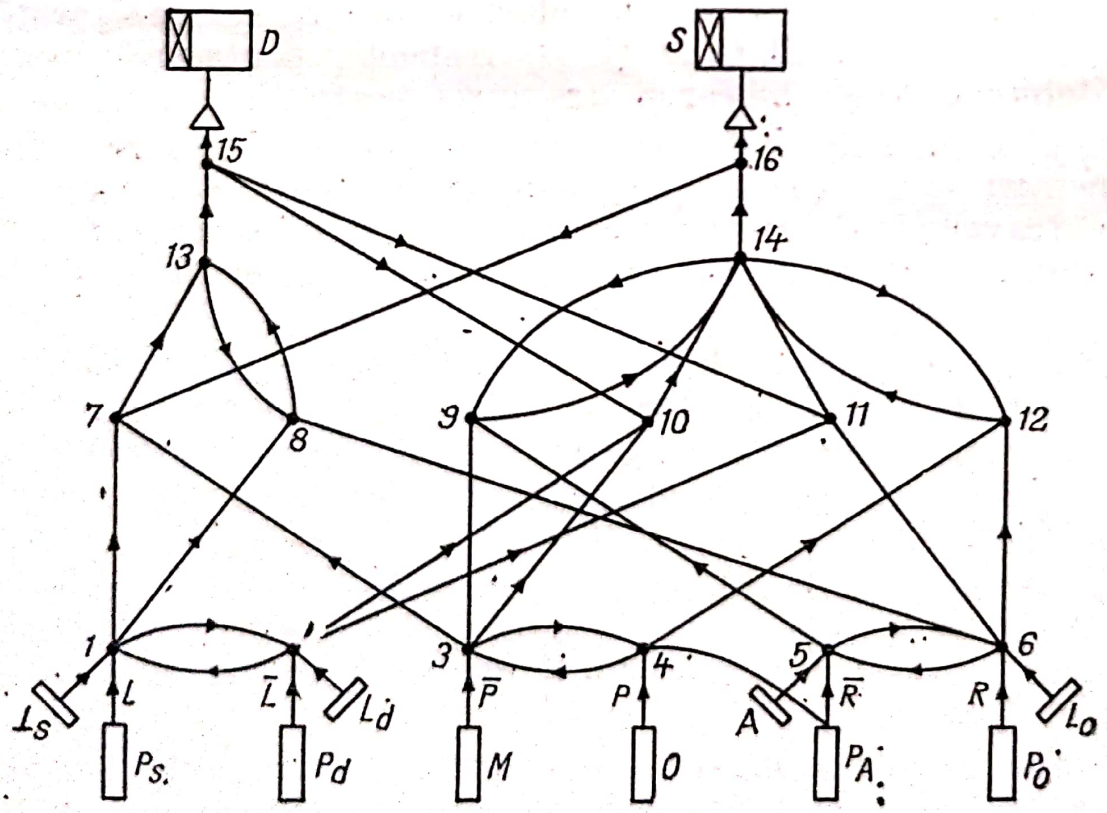


Fig. 11.32. — Schema fără contacte pentru comanda automată a unei acţionări electrice oscilante.

Scanned with OKEN Scanner

care pentru funcțiile logice ale circuitelor celor patru contactoare:

$$R = D_{1:(60,70);0:(0,1,20,21,24,30,40,42,62,100,102,120,122,124),}^{P_c P_r Lrcdb}$$

$$C = D_{1:(120,124);0:(0,1,20,21,24,30,40,42,60,62,70,100,102,122),}^{P_c P_r Lrcdb}$$

$$D = D_{1:(40,42,100,102);0:(0,1,20,21,24,30,60,62,70,120,122,124),}^{P_c P_r Lrcdb}$$

$$B = D_{1:(20,21);0:(0,1,24,30,40,42,60,62,70,100,102,120,122,124).}^{P_c P_r Lrcdb}$$

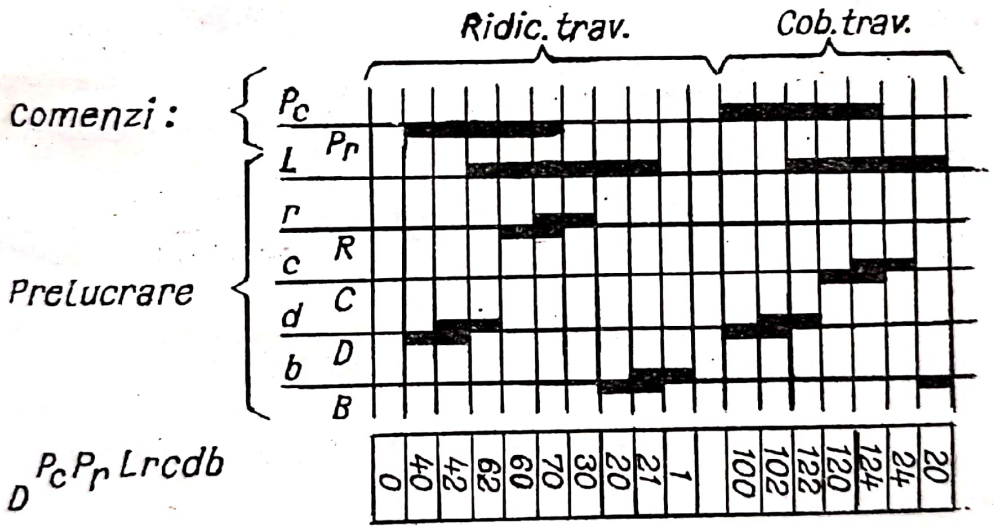


Fig. 11.33. — Diagrama de faze pentru exemplul 11.14.

Expresiile logice minimale se deduc astfel:

— pentru R:

$$\begin{array}{rl} & \begin{array}{ccc} 10 & & 1 \end{array} \\ 4 & \underline{60} \;/\; 70 \;/\; 61 & 71 \\ 100 & \underline{64} \\ & 160 \\ & \ldots \end{array}$$

$$R = D_{\underline{0}\ 1\ \underline{1000}}^{P_c P_r Lrcdb} = P_r L\bar{d} = P_r l\bar{d};$$

— pentru C :

$$\begin{array}{rl} & \overset{4}{\underline{120}} \;/\; 124 \;/\; \overset{1}{121}\ 125 \\ 10 & \underline{130} \\ 40 & \underline{160} \\ & \ldots \end{array} \qquad C = D^{P_c P_r Lrcdb}_{1\ 0\ 1\underline{0000}} = P_c L\bar{d} = P_c l\bar{d};$$

— pentru D :

$$\begin{array}{rl} & \overset{2}{\underline{40}} \;/\; 42 \;/\; \overset{1}{41}\ 43 \\ 4 & \underline{44} \\ 10 & \underline{50} \\ 100 & \underline{54} \\ & \underline{140} \\ & \ldots \end{array} \qquad D^{P_c P_r Lrcdb}_{0\ 1\ \underline{00000}} = P_r \bar{L},$$

$$\begin{array}{rl} & \overset{2}{\underline{100}} \;/\; 102 \;/\; \overset{1}{101}\ 103 \\ 4 & \underline{104} \\ 10 & \underline{110} \\ 40 & \underline{114} \\ & \underline{140} \\ & \ldots \end{array} \qquad D^{P_c P_r Lrcdb}_{1\ 0\ \underline{00000}} = P_c \bar{L};$$

rezultă deci

$$D = \bar{l}(P_r \cup P_c);$$

— pentru B :

$$\begin{array}{rl} & \overset{1}{20} \;/\; \overset{2}{21} \;/\; 22\ 23 \end{array} \qquad B = D^{P_c P_r Lrcdb}_{0\ 0\ 1\underline{0000}} = \bar{P}_c \bar{P}_r l\bar{c}.$$

Schemele cu şi fără contacte care realizează funcţiile logice cerute sînt date în figurile 11.34 şi 11.35.

Observăm necesitatea de a folosi butoane de comandă cu contacte multiple. Dacă aceasta nu e posibil, atunci se pot folosi butoane simple cu revenire, care să comande contactoare speciale ale căror contacte pot astfel îndeplini funcţiunile contactelor multiple ale butoanelor.

Scanned with OKEN Scanner

O altă soluție a acestei probleme, obținută pentru schema cu contacte prin folosirea formulei de interpolare a lui Lagrange, este dată în lucrarea [13, p. 237—240].

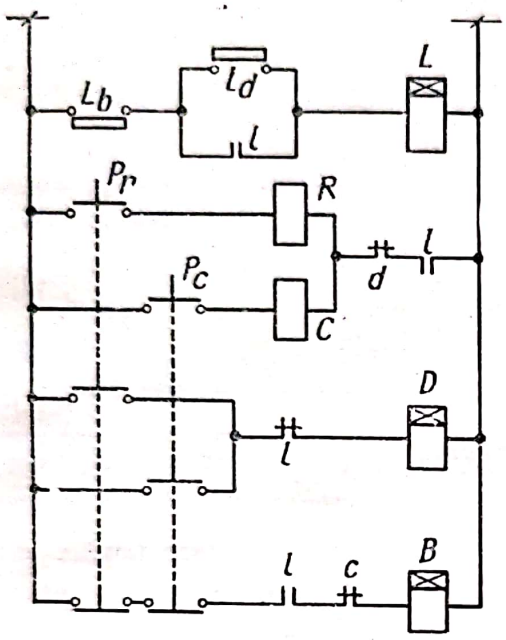


Fig. 11.34. — Schema cu contacte pentru comanda automată a mecanismului de acționare a traversei port-cuțit a unei raboteze.

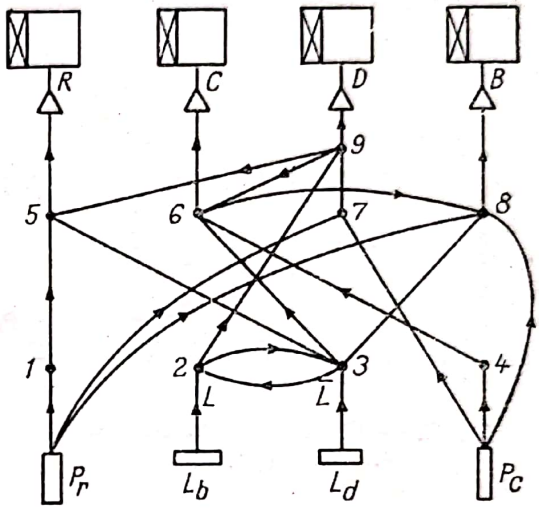


Fig. 11.35. — Schema fără contacte pentru comanda automată a mecanismului de acționare a traversei port-cuțit a unei raboteze.

Exemplul 11.15. Să se facă sinteza schemei de comandă automată a acționării unui cărucior de strung a cărui cinematică este descrisă de un ciclu pătrat, dat în figura 11.36.

Acționarea se face cu două electromotoare, al căror sens de rotație este determinat de stările a patru contactoare: X_s, X_d, Y_s și Y_d.

După darea comenzii de pornire, prin apăsarea pe butonul cu revenire P căruciorul descrie un ciclu și se oprește. Descrierea ciclului este controlată prin patru limitatoare de cursă care determină stările a două circuite de memorie.

Oprire de urgență sau de avarie este comandată printr-un alt buton cu revenire, O.

Asociind celor două circuite de memorie variabilele L pentru avansul longitudinal și T pentru avansul transversal, se poate trece la stabilirea diagramei de faze din figura 11.37.

De funcţia butonului O se va ţine seama numai în formulele finale obţinute prin sinteză. Diagrama de faze este foarte simplă şi nu necesită comentarii suplimentare. Codificarea ei ne conduce la stabilirea următorului simbol de marcare :

$$F(X_s, X_d, Y_s, Y_d) = D^{Pltx_sx_dy_sy_d}_{(10, 100, 110), (60, 64), (20, 22), (40, 41).}$$

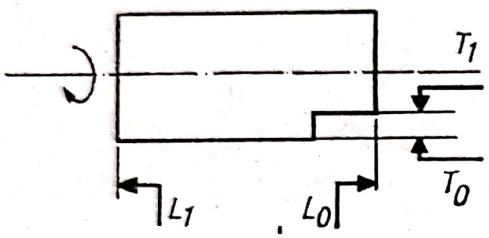


Fig. 11.36. — Ciclu pătrat la un cărucior de strung.

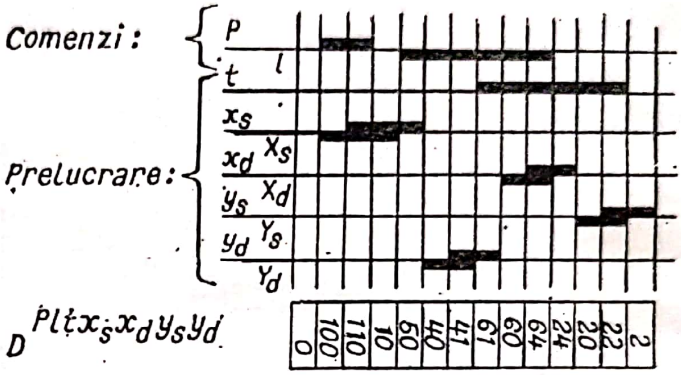


Fig. 11.37. — Diagrama de faze pentru exemplul 11.15.

Valorile binare 0 pentru cele patru funcţii logice sînt cele cuprinse în şirul

0, 2, 10, 20, 22, 24, 40, 41, 50, 60, 61, 64, 100, 110,

cu excepţia numerelor de stare date pentru fiecare funcţie logică de către simbolul de marcare.

Se deduc următoarele expresii booleene minimale :

— pentru X_s :

$$\begin{array}{rl} & \begin{array}{cccc} 100 & 1 & & 2 \end{array} \\ & \underline{10} \;/\; 110 \;/\; 11 \;\; 111 \;/\; 12 \;\; 112 \;\; 13 \;\; 113, \\ 4 & \underline{14} \\ 20 & \underline{30} \\ & \cdots \end{array}$$

$$D^{Pltx_sx_dy_sy_d}_{\underline{0}001\,\underline{0}\,\underline{0}\,\underline{0}} = \bar{l}x_s,$$

$$\begin{array}{rl} & \begin{array}{cccc} & 1 & 2 & 4 \end{array} \\ & \underline{100} \;/\; 101 \;/\; 102 \;\; 103 \;/\; 104 \;\; 105 \;\; 106 \;\; 107, \\ 10 & \underline{110} \\ 20 & \underline{120} \\ 40 & \underline{130} \\ & \underline{140} \\ & \cdots \end{array}$$

$$D^{Pltx_sx_dy_sy_d}_{\underline{1}000\,\underline{0}\,\underline{0}\,\underline{0}} = P(\bar{l});$$

rezultă :

$$X_s = (P \cup \bar{l}x)\,\overline{O} = \overline{O}\bar{l}(P \cup x_s).$$

Variabila $\bar{l}$ este redundantă și poate fi adăugată, întrucît diferența de adiacență 40 nu este obligatoriu să fie folosită pentru cuprinderea numerelor de stare conținute de simbolul de marcare X_s. Prin folosirea ei se evidențiază mai bine memoria folosită pentru realizarea funcției logice X_s;

— pentru X_d:

$$100\ \underset{\underline{160}}{\overset{4}{\underline{60}}}\ /\ 64\ /\ 62\ \overset{10}{66}\ /\ 70\ 74\ 72\ 76,$$

$$X_d = D^{Pltx_sx_dy_sy_d}_{\underline{0}11\underline{0}\ \underline{0}\ \underline{0}\ 0} = lt\overline{y}_d \cdot \overline{O};$$

— pentru Y_s:

$$100\ \underset{\underline{120}}{\overset{2}{\underline{20}}}\ /\ 22\ /\ 21\ \overset{10}{23}\ /\ 30\ 32\ 31\ 33,$$

$$Y_s = D^{Pltx_sx_dy_sy_d}_{\underline{0}01\underline{0}\ 0\ \underline{0}\ \underline{0}} = \bar{l}t\bar{x}_d\overline{O};$$

— pentru Y_d:

$$100\ \underset{\underline{140}}{\overset{1}{\underline{40}}}\ /\ 41\ /\ 42\ \overset{4}{43}\ /\ 44\ 45\ 46\ 47,$$

$$Y_d = D^{Pltx_sx_dy_sy_d}_{\underline{0}100\ \underline{0}\ \underline{0}\ \underline{0}} = l\bar{t}\bar{x}_s\overline{O}.$$

Schemele cu și fără contacte care corespund acestor formule logice sînt date în figurile 11.38 și 11.39.

Exemplul 11.16. Să se facă sinteza schemei de comandă automată a acționării unui excavator după un ciclu cinematic cubic,

format din : 1) închiderea cupei excavatorului, 2) ridicarea cupei, 3) rotaţia spre stînga a braţului excavatorului, 4) coborîrea cupei, 5) deschiderea cupei, 6) ridicarea cupei, 7) rotaţia spre dreapta a braţului excavatorului, 8) coborîrea cupei etc.

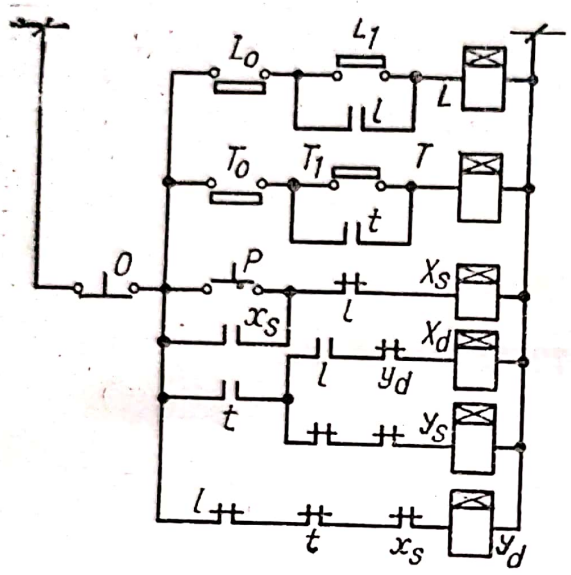


Fig. 11.38. — Schema cu contacte pentru comanda automată a unui ciclu de funcţionare la un strung automat.

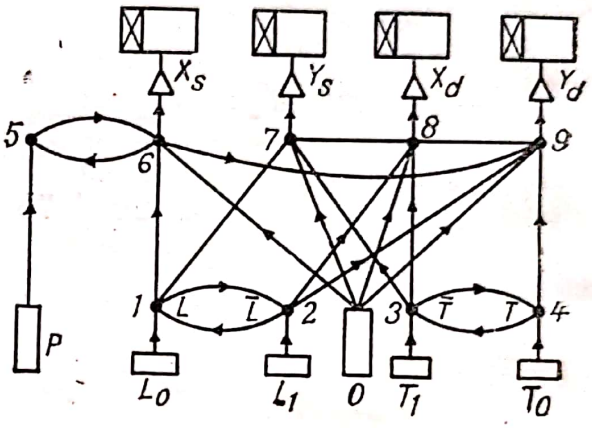


Fig. 11.39. — Schema fără contacte pentru comanda automată a unui ciclu de funcţionare la un strung automat.

Cele trei mişcări, şi anume închiderea şi deschiderea cupei, ridicarea şi coborîrea cupei şi rotaţia braţului excavatorului în plan orizontal, sînt executate cu ajutorul a trei electromotoare cu comandă reversibilă, iar poziţiile-limită sînt controlate prin limitatoare de cursă cu memorie.

Schema de comutaţie care poate îndeplini funcţiile logice cerute va avea şase variabile de ieşire, care corespund la cele şase contactoare de cuplare la reţeaua de alimentare a electromotoarelor.

Vom folosi notaţia X, Y şi Z pentru cele trei direcţii de mişcare şi vom asocia indicii r (ridicare), c (coborîre), d (dreapta), s (stînga), i (închidere) şi e (deschidere) pentru operaţiile care au loc în ciclul de lucru al excavatorului. Pentru limitatoarele de cursă vom folosi notaţia L, afectată de aceiaşi indici.

Avînd făcute aceste precizări, putem trasa diagrama de faze din figura 11.40. În această diagramă se arată secvenţele de lucru care corespund specificaţiilor date prin enunţul problemei de sinteză.

Numerele de stare rezultate prin codificarea acestei diagrame ne permit să scriem următorul simbol de marcare pentru schema de comutaţie care realizează funcţiunile cerute prin enunţ :

$$F(X_i, X_e, Y_a, Y_s, Z_r, Z_c) = D^{Pl_x l_y l_z x_i x_e y_d y_s z_r z_c}$$
$$X_i : (40, 1.000, 1.040),$$
$$X_e : (600, 620),$$
$$Y_a : (300, 310),$$
$$Y_s : (500, 504),$$
$$Z_r : (200, 202, 400, 402),$$
$$Z_c : (100, 101, 700, 701).$$

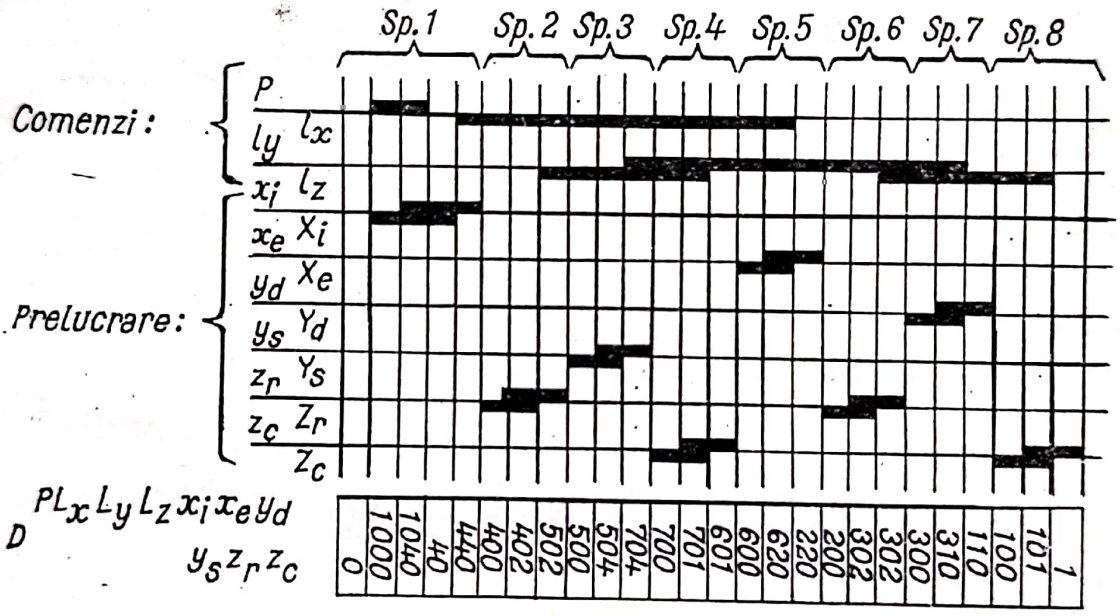


Fig. 11.40. — Diagrama de faze pentru exemplul 11.16.

Stările 0 pentru fiecare variabilă de ieşire se obţin extrăgînd stările 1, date de simbolul de marcare pentru variabila respectivă, din şirul

0, 1, 40, 100, 101, 110, 200, 202, 220, 300, 302, 310, 400, 402, 440, 500, 502, 504, 600, 601, 620, 700, 701, 704, 1.000, 1.040.

Folosind informaţiile de mai sus, deducem următoarele expresii booleene minimale pentru funcţiile logice ale variabilelor de ieşire :

— pentru X_i :

$$\begin{array}{l} \overset{1.000}{} \qquad \overset{1}{} \qquad \overset{2}{} \\ 4\,\underline{40} \;/\; 1.040 \;/\; 41\; 1.041 \;/\; 42 \;\; 1.042 \;\; 43 \;\; 1.043, \\ 10\,\underline{44} \qquad 100\,\underline{\qquad} \\ \quad\; \underline{50} \qquad\quad \underline{140} \\ 20\,\underline{54} \qquad\quad \ldots \\ \quad\; \underline{60} \qquad 200\,\underline{\qquad} \qquad\qquad D_{\underline{00}}^{Pl_x\, l_y\, l_z\, x_i\, x_e\, \bar{y}_d\, y_s\, z_r\, z_c} = \bar{l}_x\, x_i, \\ \quad\; .. \qquad\quad \underline{240} \\ \qquad\qquad\quad \ldots \end{array}$$

$$\begin{array}{l} \overset{1}{} \qquad \overset{2}{} \qquad\quad \overset{4}{} \\ 10\,\underline{1.000} \;/\; 1.001 \;/\; 1.002 \;\; 1.003 \;/\; 1.004 \;\; 1.005 \;\; 1.006 \;\; 1.007 \\ 20\,\underline{1.010} \qquad 100\,\underline{\qquad} \\ \quad\; \underline{1.020} \qquad\quad \underline{1.100} \\ 40\,\underline{1.030} \qquad\quad \ldots\ldots \\ \quad\; \underline{1.040} \\ \quad \ldots\ldots \qquad 200\,\underline{\qquad} \\ \qquad\qquad\quad \underline{1.200} \\ \qquad\qquad\quad \ldots\ldots \\ \\ \qquad\qquad 400\,\underline{\qquad} \qquad\qquad D_{\underline{10}\;\underline{0}\;\underline{0}\;\underline{0}\;\underline{0}\;\underline{0}\;\underline{0}\;\underline{0}\;\underline{0}}^{Pl_x\, l_y\, l_z\, x_i\, x_e\, y_d\, y_s\, z_r\, z_c} = P(\bar{l}_x); \\ \qquad\qquad\quad \underline{1.400} \\ \qquad\qquad\quad \ldots\ldots \end{array}$$

rezultă

$$X_i = \bar{l}_x\, \overline{O}\, (P \cup x_i);$$

— pentru X_e :

$$\begin{array}{l} \overset{20}{} \qquad \overset{2}{} \qquad\quad \overset{4}{} \\ \quad 10\,\underline{600} \;/\; 620 \;/\; 602 \;\; 622 \;/\; 604 \;\; 624 \;\; 606 \;\; 626, \\ \quad\; \underline{610} \\ \quad 40\,\underline{640} \\ 1.000\,\underline{650} \qquad X_e = D_{\underline{01}\;1\;0\;\underline{0}\;\underline{0}\;\underline{0}\;\underline{0}\;\underline{0}\;0}^{Pl_x\, l_y\, l_z\, x_i\, x_e\, y_d\, y_s\, z_r\, z_c} = l_x\, l_y\, \bar{l}_z\, \bar{z}_c\, \overline{O}; \\ \quad\; \underline{1.600} \\ \quad \ldots\ldots \end{array}$$

— pentru Y_d :

$$\begin{array}{r} \\ 20\,\dfrac{300}{320} \\ 40\,\dfrac{340}{360} \\ 1.000\,\dfrac{1.300}{\ldots\ldots} \end{array} \quad \begin{array}{c} \overset{10}{/\ 310} \\ \\ \\ \\ \end{array} \quad \begin{array}{c} \overset{1}{/\ 301\ \ 311} \\ \\ \\ \\ \end{array} \quad \begin{array}{c} \overset{4}{/\ 304\ 314\ 305\ 315}, \\ \\ \\ \\ \end{array}$$

$$Y_d = D^{Pl_x\, l_y\, l_z\, x_i\, x_e\, y_d\, y_s\, z_r\, z_c}_{\underline{0}0\ 1\ 1\ \underline{0}\ \underline{0}\ \underline{0}\ \ \underline{0}\ 0\ \underline{0}} = \bar{l}_x\, l_y\, l_z\, \bar{z}_r\, \overline{O}\,;$$

— pentru Y_s :

$$\begin{array}{r} \\ 20\,\dfrac{500}{520} \\ 40\,\dfrac{540}{560} \\ 1.000\,\dfrac{1.500}{\ldots\ldots} \end{array} \quad \begin{array}{c} \overset{4}{/\ 504} \\ \\ \\ \\ \end{array} \quad \begin{array}{c} \overset{1}{/\ 501\ \ 505} \\ \\ \\ \\ \end{array} \quad \begin{array}{c} \overset{10}{/\ 510\ 514\ 511\ 515}, \\ \\ \\ \\ \end{array}$$

$$Y_s = D^{Pl_x\, l_y\, l_z\, x_i\, x_e\, y_d\, y_s\, z_r\, z_c}_{\underline{0}1\ 0\ 1\ \underline{0}\ \underline{0}\ \underline{0}\ \ \underline{0}\ 0\ \underline{0}} = \bar{l}_x\, l_y\, l_z\, z_r\, \overline{O}\,;$$

— pentru Z_r :

$$\begin{array}{r} \\ 10\,\dfrac{200}{210} \\ 40\,\dfrac{240}{230} \\ 1.000\,\dfrac{1.200}{\ldots\ldots} \end{array} \quad \begin{array}{c} \overset{2}{/\ 202} \\ \\ \\ \\ \end{array} \quad \begin{array}{c} \overset{1}{/\ 201\ \ 203} \\ \\ \\ \\ \end{array} \quad \begin{array}{c} \overset{4}{/\ 204\ 206\ 205\ 207}, \\ \\ \\ \\ \end{array}$$

$$D^{Pl_x\, l_y\, l_z\, x_i\, x_e\, y_d\, y_s\, z_r\, z_c}_{\underline{0}0\ 1\ 0\ \underline{0}\ 0\ \underline{0}\ \ \underline{0}\ \ \underline{0}\ \ \underline{0}} = \bar{l}_x\, l_y\, \bar{l}_z\, \bar{x}_e,$$

$$\begin{array}{r} \\ 10\,\dfrac{400}{410} \\ 20\,\dfrac{420}{430} \\ 1.000\,\dfrac{1.400}{\ldots\ldots} \end{array} \quad \begin{array}{c} \overset{2}{/\ 402} \\ \\ \\ \\ \end{array} \quad \begin{array}{c} \overset{1}{/\ 401\ \ 403} \\ \\ \\ \\ \end{array} \quad \begin{array}{c} \overset{4}{/\ 404\ \ 604\ \ 405\ \ 407}, \\ \\ \\ \\ \end{array}$$

$$D^{Pl_x\, l_y\, l_z\, x_i\, x_e\, y_d\, y_s\, z_r\, z_c}_{\underline{0}1\ 0\ 0\ 0\ \underline{0}\ \underline{0}\ \ \underline{0}\ \ \underline{0}\ \ \underline{0}} = l_x\, \bar{l}_y\, \bar{l}_z\, \bar{x}_i\,;$$

rezultă deci

$$Z_r = \bar{l}_z (\bar{l}_x\, l_y\, \bar{x}_e \cup l_x\, \bar{l}_y\, x_i)\, \overline{O};$$

— pentru Z_c:

$$\begin{array}{l} \quad\quad\quad\

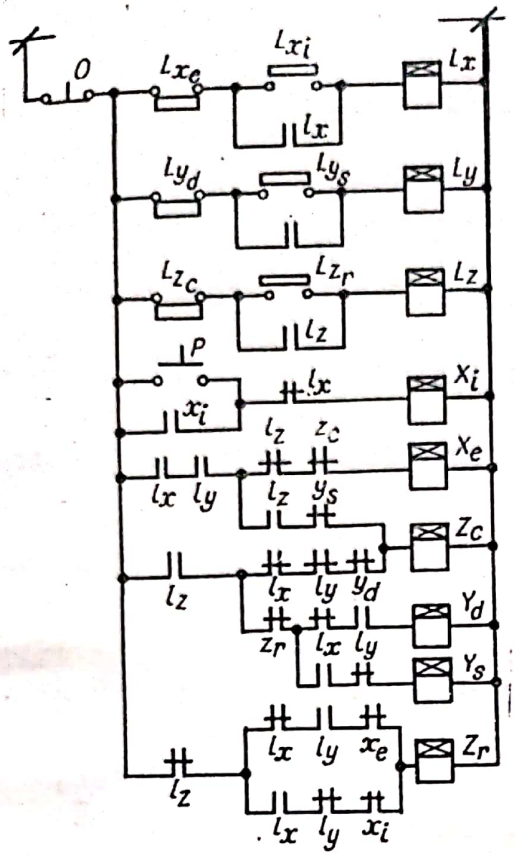


Fig. 11.41. — Schema cu contacte pentru comanda automată a unui excavator.

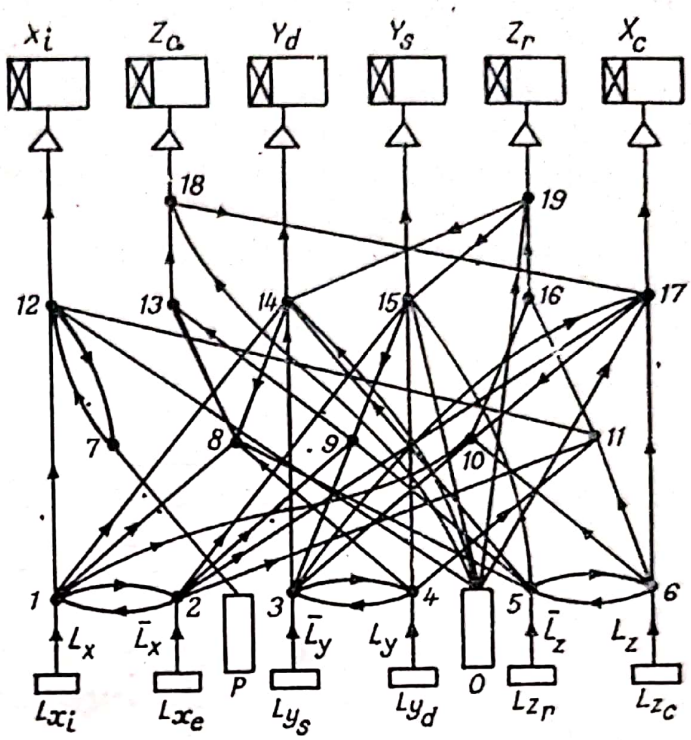


Fig. 11.42. — Schema modulară UNILOG pentru comanda automată a unui excavator.

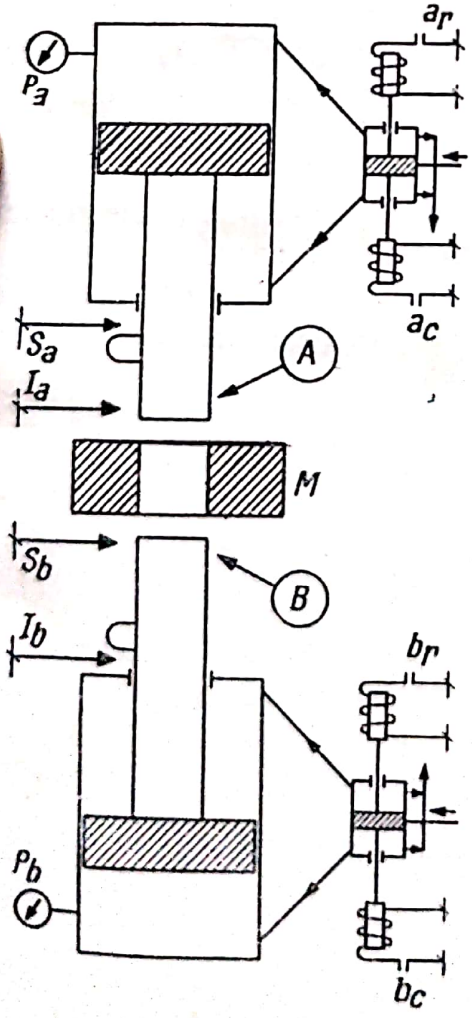


Fig. 11.43. — Acționarea unei prese cu dublu efect.

Poziţiile culiselor A şi B ale presei sînt controlate prin limitatoare cu memorie, notate prin S_a, I_a şi S_b, I_b, iar forţa lor de presare este controlată de traductoarele de presiune cu contact P_a şi P_b.

Comanda punerii în funcţiune a presei se dă cu ajutorul unui buton P cu revenire.

Butonul de oprire are şi rol de întrerupător de avarie sau de comandă a opririi de urgenţă.

Ciclul de lucru al presei cuprinde următoarele faze :

1) se presupune că matriţa e alimentată. În acest caz are loc coborîrea culisei A pînă la nivelul matriţei M;

2) urcarea culisei B pînă în momentul cînd presiunile exercitate de cele duoă culise asupra materialului din matriţă ating valorile prescrise;

3) ridicarea culisei A pînă la nivelul limită superior;

4) ridicarea culisei B care împinge comprimatul afară;

5) coborîrea culisei B.

Urmează apoi îndepărtarea comprimatului, alimentarea matriţei şi începutul unui nou ciclu de funcţionare. Aceste operaţii se presupun că se fac în timpul cît durează coborîrea culisei B.

Considerăm drept variabile primare variabilele asociate ieşirii celor două memorii L_a şi L_b, cele asociate traductoarelor de presiune cu contact P_a şi P_b şi variabila asociată butonului de pornire P.

Variabilele de ieşire şi cele secundare sînt asociate celor patru contactoare care servesc la acţionarea electromotoarelor presei.

Cu aceste precizări putem trece la trasarea diagramei de faze din figura 11.44, în care sînt evidenţiate secvenţele de lucru care corespund specificaţiilor din enunţ.

Simbolul de marcare, rezultat din citirea diagramei de faze codificate, este

$$F(A_r, A_c, B_r, B_c) = D^{Pl_a\, l_b\, p_a\, p_b\, a_r\, a_c\, b_r\, b_c}_{(10,\ 210,\ 230,\ 250,\ 260,\ 270),\ (4,\ 400,\ 404),\ (2,\ 10{,}12,\ 200,\ 202,\ 222,\ 242),\ (100,\ 101).}$$

Stările 0 pentru fiecare variabilă de ieşire se obţin extrăgînd stările 1, date de simbolul de marcare pentru variabila respectivă, din şirul

0, 1,2,4,10, 12, 100, 101, 102, 200, 202, 204, 210, 222, 230, 242, 250, 260, 262, 270, 400, 404.

Expresiile booleene minimale care rezultă sînt următoarele:

— pentru A_r:

$$\begin{array}{l} \quad\quad\quad\

rezultă

$$A_r = \overline{b}_r (a_r \cup p_a) \overline{O};$$

— pentru A_c:

400 1 2
/ 404 / 5 405 / 6 406 7 407

10 4/14
20 24
40 34/44 100 104 $D^{Pl_a\, l_b\, p_a\, p_b\, a_r\, a_c\, b_r\, b_c}_{\underline{0}0\ \underline{0}\ \underline{0}\ \underline{0}\ \underline{0}\ 1\ \underline{0}\ \underline{0}} = \overline{l}_a\, a_c,$
... ...

1 2 4
/ 401 / 402 403 / 404 405 406 407

10 400/410
20 420 100 500
40 430/440 ...
... 200 600 $D^{Pl_a\, l_b\, p_a\, p_b\, a_r\, a_c\, b_r\, b_c}_{\underline{1}0\ \underline{0}\ \underline{0}\ \underline{0}\ \underline{0}\ \underline{0}\ \underline{0}\ \underline{0}} = P(\overline{l}_a);$
...

rezultă deci

$$A_c = \overline{O}\overline{l}_a (P \cup a_c);$$

— pentru B_r:

10 1 4
2/ 12 / 3 13 / 6 16 7 17,

20 22/42
40
400 62/402 $D^{Pl_a l_b p_a p_b a_r a_c b_r b_c}_{\underline{0}0\ 0\ \underline{0}\ \underline{0}\ \underline{0}\ \underline{0}\ 1\ \underline{0}} = \overline{l}_b b_r \overline{l}_a,$
...

$$\begin{array}{rl} & \begin{array}{cccc} 1 & 2 & & 4 \end{array} \\ 20 & \underline{10} \;/\; 11 \;/\; 12\;13 \;/\; 14\;15\;16\;17, \\ 40 & \underline{30} \\ & \underline{50} \\ 100 & \underline{70} \quad 400 \quad \underline{\quad\quad} \\ & \underline{110} \qquad\quad 410 \\ & \ldots \qquad\quad \ldots \end{array} \qquad D^{Pl_a l_b p_a p_b a_r a_c b_r b_c}_{\underline{0}0\,\underline{0}\,\underline{0}\,\underline{0}\;1\;\underline{0}\,\underline{0}\,\underline{0}} = \bar{l}_a a_r,$$

$$\begin{array}{rl} & \begin{array}{cccc} 1 & 2 & & 40 \end{array} \\ 100 & \underline{200} \;/\; 201 \;/\; 202\;203 \;/\; 240\;241\;242\;243, \\ 400 & \underline{300} \\ & \underline{600} \\ & 700 \end{array} \qquad D^{Pl_a l_b p_a p_b a_r a_c b_r b_c}_{\underline{0}1\,\underline{0}\,\underline{0}\;0\;0\;0\,\underline{0}\,\underline{0}} = l_a \bar{p}_b \bar{a}_r \bar{a}_c;$$

rezultă deci

$$B_r = \overline{O}\,(\bar{l}_a \bar{l}_b b_r \cup \bar{l}_a a_r \cup l_a \bar{p}_b \bar{a}_r \bar{a}_c);$$

— pentru B_c:

$$\begin{array}{rl} & \begin{array}{cccc} 1 & 4 & & 10 \end{array} \\ 20 & \underline{100} \;/\; 101 \;/\; 104\;105 \;/\; 110\;111\;114\;115, \\ & \underline{120} \quad 40 \quad \underline{\quad\quad} \qquad 200 \quad \underline{\quad\quad} \quad 400 \quad \underline{\quad\quad} \\ & \ldots \qquad\quad \ldots \qquad\quad \ldots \qquad\quad \ldots \end{array}$$

$$B_c = D^{Pl_a l_b p_a p_b a_r a_c b_r b_c}_{\underline{0}\underline{0}\;1\;\underline{0}\,\underline{0}\,\underline{0}\,\underline{0}\;0\;\underline{0}} = l_b \bar{b}_r.$$

Schemele cu şi fără contacte care corespund expresiilor logice deduse sînt date în figurile 11.45 şi 11.46.

Exemplul 11.18. Să se facă sinteza schemei de comandă automată a unei maşini automate de găurit. Piesa P care trebuie perforată, este antrenată într-o mişcare de translaţie cu un mecanism bielă-manivelă, acţionat de un motor X. Poziţiile extreme sînt reperate prin limitatoarele de fine de cursă A şi B.

Rotaţia burghiului este cauzată de motorul Y, iar poziţia burghiului după perforarea piesei P şi revenirea la starea iniţială este reperată prin limitatorul de fine de cursă C. Mişcarea de translaţie a burghiului este cauzată de motorul Z.

Comanda de pornire este dată prin butonul cu revenire M.

Scanned with OKEN Scanner

Variabilele primare ale schemei secvențiale de comutație care asigură funcționarea automată a mașinii de găurit vor fi cele asociate limitatoarelor de fine de cursă A, B și C și butonului de comandă M.

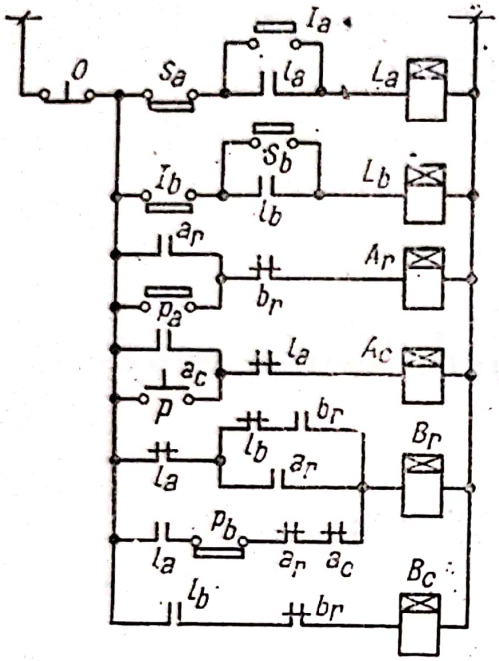


Fig. 11.45. — Schema cu contacte pentru comanda automată a acționării unei prese cu dublu efect.

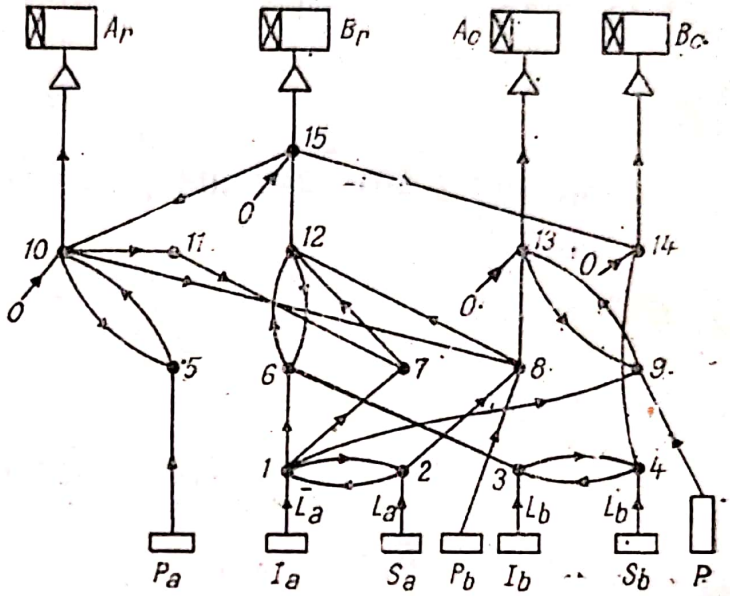


Fig. 11.46. — Schema modulară UNILOG pentru comanda automată a acționării unei prese cu dublu efect.

Drept variabile de ieșire se consideră inițial numai cele asociate contactoarelor X, Y și Z care cuplează electromotoarele de acționare la rețeaua electrică. Rezultă astfel raportul de stare

$$\frac{ABCM}{XYZ} = \frac{n}{XYZ}.$$

Starea inițială a schemei corespunde la $B = C = l$ și $A = M = X = Y = Z = 0$, fiind redată simbolic prin raportul 6/000.

Secvența de lucru a schemei, care corespunde programului de lucru a mașinii, se deduce astfel: prin apăsarea butonului de comandă M, piesa P se pune în mișcare ($X = 1$), părăsind reperul B ($B = 0$), astfel încît schema trece prin stările: 7/100→3/100.

Dezacționarea butonului M ($M = 0$) nu modifică starea schemei numai dacă piesa P începe să se deplaseze, în acest caz schema trecînd în starea 2/100. Cînd piesa P atinge reperul A ($A = 1$), ea este oprită ($X = 0$) și simultan începe avansul burghiului ($Z = 1$), prin aceasta burghiul părăsind reperul C ($C = 0$) și începînd rotația lui ($Y = 1$). Schema trece astfel prin stările 12/001→10/011. Urmează apoi perforarea piesei P și, în momentul cînd burghiul s-a întors la reperul C ($C = 1$), are loc simultan oprirea burghiului ($Z = Y = 0$) și punerea în mișcare a piesei P ($X = 1$) care trebuie readusă la reperul inițial B.

Întrucît noua stare (12), corespunde altor valori ale variabilei de ieșire, se impune introducerea variabilei de ieșire S și a celei secundare s asociate ei, prin care să discriminăm cele două stări. Aceasta se face prin stările 10/(1)011 → 30/(1)011 → 32/(1)100 → 22/(1)100.

Odată atins reperul B ($B = 1$), este oprită deplasarea piesei P ($X = 0$) și se ajunge la starea inițială. Astfel avem 26/(0)100 → 6.

Un nou ciclu de lucru poate fi comandat printr-o nouă apăsare a butonului de comandă M.

Sintetizăm stările stabilite în următoarea secvență de lucru a schemei:

$$\frac{6}{(0)\,000} \xrightarrow{M=1} \frac{7}{(0)\,100} \xrightarrow{B=0} \frac{3}{(0)\,100} \xrightarrow{M=0} \frac{2}{(0)\,100} \xrightarrow{A=1} \frac{12}{(0)\,001} \rightarrow$$

$$\xrightarrow{C=0} \frac{10}{(1)\,011} \rightarrow \frac{30}{(1)\,011} \xrightarrow{C=1} \frac{32}{(1)\,100} \xrightarrow{A=0} \frac{22}{(1)\,100} \rightarrow$$

$$\xrightarrow{B=1} \frac{26}{(0)\,000} \rightarrow (6).$$

Observăm că variabilele atașate limitatoarelor de fine de cursă joacă rolul unor pseudovariabile secundare.

Pentru a reda mai bine aspectul fizic al secvenței de lucru a schemei, în figura 11.47 — s-a dat și diagrama de faze corespunzătoare.

Scanned with OKEN Scanner

Stările secvenței de lucru a schemei definesc următorul simbol de marcare:

$$F(X, Y, Z, S) = D^{sABCM}_{\substack{(2,\ 3,\ 7,\ 22,\ 26,\ 32,\ 6^\circ,\ 10^\circ,\ 12^\circ,\ 30^\circ),\\ (10,\ 30,\ 2^\circ,\ 3^\circ,\ 6^\circ,\ 7^\circ,\ 12^\circ,\ 22^\circ,\ 26^\circ,\ 32^\circ),\\ (10,\ 12,\ 30,\ 2^\circ,\ 3^\circ,\ 6^\circ,\ 7^\circ,\ 22^\circ,\ 26^\circ,\ 32^\circ)\\ (10,\ 22,\ 30,\ 32,\ 2^\circ,\ 3^\circ,\ 6^\circ,\ 7^\circ,\ 12^\circ,\ 26^\circ).}}$$

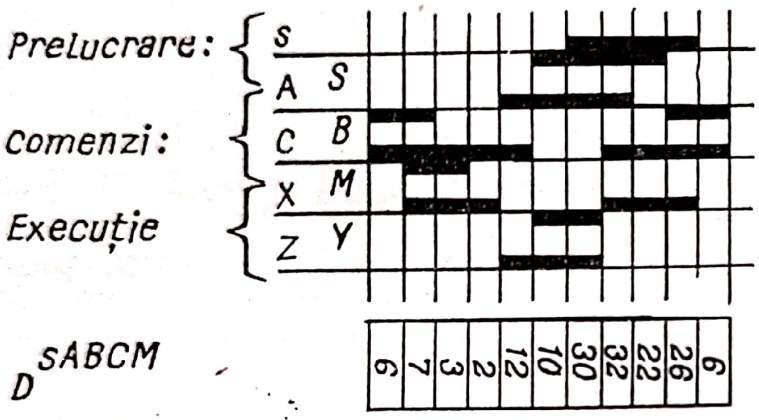


Fig. 11.47. — Diagrama de faze pentru exemplul 11.18.

Aplicînd metoda simbolică de minimizare, se stabilesc următoarele expresii minimale ale celor patru funcții logice, definite de simbolul de marcare următor:

— pentru X:

$$\underline{2}\ /\ 3\ /\ 22\ 23\ /\ 0\ 1\ 20\ 21 \qquad D^{sABCM}_{\underline{0}\,00\underline{1}\,\underline{0}} = \bar{A}\bar{B},$$

$$20\ \underline{2}\overline{7}\ /\ 5\ /\ 3\ 1\ /\ 17\ 15\ 13\ 11, \qquad D^{sABCM}_{\underline{00111}} = M,$$

$$26\ /\ 36\ /\ 22\ 32\ /\ 27\ 37\ 23\ 33, \qquad D^{sABCM}_{1\,\underline{01}\,1\,\underline{0}} = sC;$$

rezultă deci

$$X = M \cup \bar{A}\bar{B} \cup sC;$$

— pentru Y:

$$10\ \frac{10}{0}\ \overset{20}{/}\ 30\ \overset{1}{/}\ 11\ 31\ \overset{4}{/}\ 14\ 34\ 15\ 35. \qquad Y = D^{sABCM}_{\underline{0\,1}\,0\,0\,\underline{0}} = \overline{C};$$

— pentru Z:

$$10\ \overset{2}{/}\ 12\ \overset{1}{/}\ 11\ 13\ \overset{4}{/}\ 14\ 16\ 15\ 17, \qquad D^{sABCM}_{0\,1\,\underline{0\,0\,0}} = \bar{s}A,$$

$$20\ \frac{30}{10}\ \overset{1}{/}\ 31\ \overset{4}{/}\ 34\ 35\ \overset{10}{/}\ 20\ 21\ 24\ 25, \qquad D^{sABCM}_{\underline{1\,1}\,0\,0\,\underline{0}} = \overline{C};$$

rezultă deci

$$Z = \bar{s}A \cup \overline{C};$$

— pentru S:

$$10\ \frac{10}{0}\ \overset{20}{/}\ 30\ \overset{1}{/}\ 11\ 31\ \overset{4}{/}\ 14\ 34\ 15\ 35, \qquad D^{sABCM}_{\underline{0\,1}\,0\,0\,\underline{0}} = \overline{C}$$

$$22\ \overset{10}{/}\ 32\ \overset{1}{/}\ 23\ 33\ \overset{2}{/}\ 20\ 30\ 21\ 31, \qquad D^{sABCM}_{1\,\underline{0}\,0\,\underline{1\,0}} = s\bar{B};$$

rezultă deci

$$S = \overline{C} \cup s\bar{B}.$$

Aceste formule permit structurarea schemei cu contacte date în figura 11.48 *a*.

Pentru a găsi schema echivalentă cu punți, ne folosim de regulile transformării din stea în triunghi, expuse în capitolul 9.

Pentru aceasta considerăm funcțiile logice specifice

$$\Delta_x = M \cup \overline{A}\overline{B} \cup sC;\ \Delta_y = \overline{C};\ \Delta_z = \bar{s}A \text{ și } \Delta_S = s\bar{B}.$$

Aceste funcții logice cuprind în ansamblu toate numerele de stare care definesc mărimile de ieșire ale schemei, iar fiecare în parte cuprinde numerele de stare specifice variabilei de ieșire pe care o definesc.

O legătură în punte posibilă este dată de

$$\Delta_{zS} = \overline{C}.$$

Astfel, întrucît Δ_z şi Δ_S nu au numere de stare comune, Δ_{zS} va fi definit de numerele de stare comune variabilelor Z şi S, dar necuprinse de Δ_z şi Δ_S, adică de numerele de stare care definesc variabila $\overline{C}$.

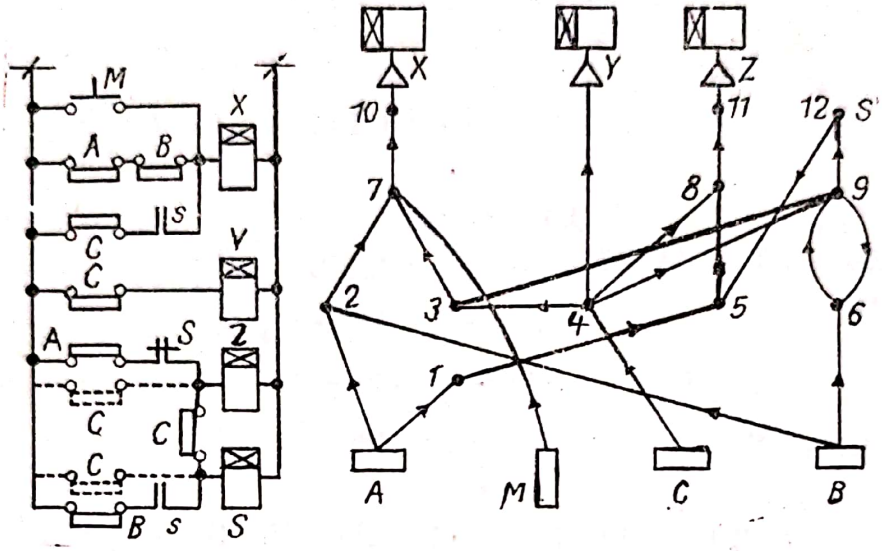


Fig. 11.48. — Schemele de comutaţie a) dinamică şi b) statică pentru comanda automată a unei maşini de găurit.

Pe calea acestei transformări, cele două contacte $\overline{C}$, reprezentate prin linii întrerupte în schema din fig. 11.48*a*, au fost înlocuite printr-un contact unic $\overline{C}$, conectat pe puntea de legătură dintre Z şi S.

Pentru structurarea schemei modulare UNILOG, punem cele patru funcţii logice sub forma

$$X = \overline{\overline{M \cup \overline{A}\,\overline{B} \cup sC}} = \overline{\overline{M}\,(A \cup B)\,(\overline{s} \cup \overline{C})} = \overline{\overline{M \cup (A \cup B) \cup (\overline{s} \cup \overline{C})}} =$$

$$= \overline{M \top (A \top B) \top (\overline{s} \top \overline{C})},\quad Y = \overline{C},$$

$$Z = \overline{\overline{sA \cup \overline{C}}} = (s \cup \overline{A})\,C = \overline{\overline{(s \cup \overline{A}) \cup \overline{C}}} = (s \top \overline{A}) \top \overline{C},$$

$$S = \overline{\overline{C \cup sB}} = C\,(\overline{s} \cup B) = \overline{\overline{C} \cup (\overline{s} \cup B)} = \overline{C} \top (\overline{s} \top B).$$

Schema modulară care corespunde acestor formule este reprezentată în figura 11.48*b*. Ea a fost verificată experimental cu ajutorul unui simulator logic și a corespuns specificațiilor date prin enunțul problemei de sinteză.

O problemă similară este rezolvată și în [84, p. 276] prin alte metode. Rezultatele obținute prin metoda simbolică de sinteză ne conduc la soluții mai avantajoase economic, pe o cale de rezolvare, credem, mai simplă. Astfel schema echivalentă conține numai 11 contacte în loc de 13, cum se arată în lucrarea citată.

Exemplul 11.19. Să se facă sinteza schemei de comandă automată a unei acționări electrohidraulice (fig. 11.49) pentru o mișcare du-te-vino controlată printr-un singur limitator de cursă.

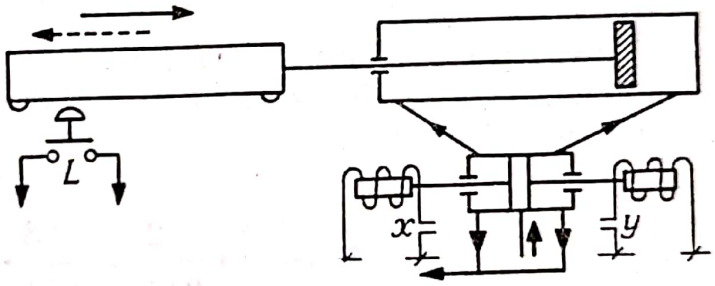


Fig. 11.49. — Acționare electrohidraulică.

Considerăm drept variabile primare variabila L, asociată contactului limitatorului de cursă, și variabila P, asociată butonului de pornire.

Variabilele de ieșire și cele secundare le asociem contactoarelor X și Y, ale căror contacte x și y controlează alimentarea electromagneților care acționează sertarul amplificatorului hidraulic ce comandă servomotorul hidraulic de acționare.

Pentru memorarea sensului de mișcare este nevoie de introducerea unei variabile auxiliare de ieșire A.

Cu aceste precizări, raportul de stare al schemei va fi

$$\frac{PL(xya)}{XYA} = \frac{n}{XYA}.$$

Secvențele de lucru ale schemei se stabilesc făcînd următorul raționament: dacă în starea de repaus (0) se acționează butonul P, schema trece în starea (20), în care este acționat contactorul X, prin

anclanşarea contactelor căruia schema trece în starea (24). Dezacţionînd P, schema trece în starea (4):

$$(0) \xrightarrow{P=1} \frac{20}{100} \rightarrow \frac{24}{100} \xrightarrow{P=0} \frac{4}{100}.$$

La atingerea limitatorului L, contactorul X devine dezexcitat şi contactorul Y excitat, astfel încît schema trece prin stările

$$(4) \xrightarrow{L=1} \frac{14}{000} \rightarrow \frac{10}{010} \rightarrow \frac{12}{010}.$$

Dezacţionarea lui L nu modifică starea lui X, dar o modifică pe cea a lui A. Prin aceasta este memorat sensul de mişcare spre stînga. Astfel avem

$$(12) \xrightarrow{L=0} \frac{2}{011} \rightarrow \frac{3}{011}.$$

O nouă acţionare a lui L trece schema prin stările

$$(3) \xrightarrow{L=1} \frac{13}{001} \rightarrow \frac{11}{101} \rightarrow \frac{15}{101},$$

prin care se inversează sensul de mişcare.

În fine, o nouă dezacţionare a lui L readuce schema în starea iniţială:

$$(15) \xrightarrow{L=0} \frac{5}{100} \rightarrow \frac{4}{100}.$$

Sub formă grafică, secvenţele de lucru ale schemei se prezintă ca în diagrama de faze dată în figura 11.50.

Scanned with OKEN Scanner

Folosind numerele de stare definite prin stările de mai sus sau din diagrama de faze codificată, se stabilește următorul simbol de marcare pentru funcțiile logice ale schemei sintetizate:

$$F(X, Y, A) = D^{PLxya}_{(4, 5, 11, 15, 20, 24), (2, 3, 10, 12), (2, 3, 11, 13, 15).}$$

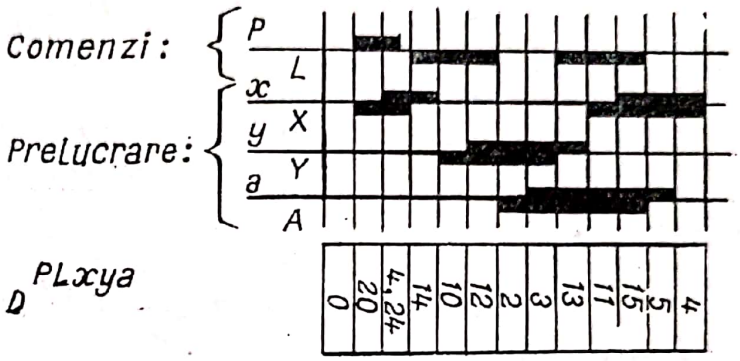


Fig. 11.50. — Diagrama de faze pentru exemplul 11.13.

Stările 0 se extrag din șirul

0, 2, 3, 4, 5, 10, 11, 12, 13, 14, 15, 20, 24.

Rezultă următoarele expresii logice minimale:
— pentru X:

$$\begin{array}{rl} & \begin{array}{cc} 1 & 20 \end{array} \\ 2 & \begin{array}{c} \underline{4} \\ 6 \end{array} \;/\; 5 \;/\; 24\ 25 \end{array} \qquad D^{PLxya}_{\underline{0}\,0\,1\underline{00}} = \overline{L}x,$$

$$\begin{array}{rl} & \begin{array}{cc} 4 & 20 \end{array} \\ 10 & \begin{array}{c} \underline{11} \\ 1 \end{array} \;/\; 15 \;/\; 31\ 35 \end{array} \qquad D^{PLxya}_{\underline{010}\,0\,1} = \overline{y}a,$$

$$\begin{array}{rl} & \begin{array}{cc} 1 & 2 \end{array} \\ 4 & \underline{20} \;/\; 21 \;/\; 22\ 23 \\ 10 & \underline{24} \\ & 30 \\ & 34 \end{array} \qquad D^{PLxya}_{\underline{1000}\,0} = P;$$

rezultă deci

$$X = \overline{O}\,(P \cup \overline{L}x \cup \overline{y}a);$$

— pentru Y :

$$20\ \underset{22}{\overset{2}{\underline{2}}}\ \overset{1}{/}\ 3\ \overset{4}{/}\ 6\ 7 \qquad D^{PLxya}_{\underline{0}\,0\,0\,\underline{1}\,\underline{0}} = \overline{L}y,$$

$$\underline{10}\ \overset{2}{/}\ 12\ \overset{20}{/}\ 30\ 32 \qquad D^{PLxya}_{\underline{0}\,1\,\underline{0}\,0\,\underline{0}} = L\bar{x}\bar{a},$$

$$20\ \underset{22}{\overset{2}{\underline{2}}}\ \overset{10}{/}\ 12\ \overset{4}{/}\ 6\ 16 \qquad D^{PLxya}_{\underline{000}\,1\,0} = (y\bar{a});$$

rezultă deci

$$= Y[\overline{L}y \cup L\bar{x}\bar{a} \cup (y\bar{a})]\,\overline{O};$$

— pentru A :

$$20\ \underset{22}{\overset{2}{\underline{2}}}\ \overset{1}{/}\ 3\ \overset{4}{/}\ 6\ 7 \qquad D^{PLxya}_{\underline{0}\,0\,0\,\underline{1}\,\underline{0}} = \overline{L}y,$$

$$20\ \underset{31}{\overset{11}{\underline{1}}}\ \overset{2}{/}\ 13\ \overset{4}{/}\ 15\ 17 \qquad D^{PLxya}_{\underline{0}\,1\,\underline{00}\,1} = La,$$

$$20\ \underset{23}{\overset{3}{\underline{3}}}\ \overset{10}{/}\ 13\ \overset{2}{/}\ 1\ 11 \qquad D^{PLxya}_{\underline{000}\,\underline{1}\,1} = (\bar{x}a);$$

rezultă deci

$$A = [Ly \cup La \cup (\bar{x}a)]\,\overline{O}.$$

Schema cu contacte care corespunde acestor formule este dată în figura 11.51, termenii redundanți necesari pentru eliminarea hazardului de continuitate fiind reprezentați prin linii întrerupte.

Schema modulară UNILOG este dată în figura 11.52.

Exemplul 11.20. Să se facă sinteza schemei sincrone de comandă automată a unui transportor cu bandă format din trei tronsoane.

Comanda de pornire-oprire este dată printr-o memorie comandată la intrare prin butoanele cu revenire M și O.

Pentru memorarea stărilor schemei în pauzele dintre impulsurile sincrone provenite de la generatorul de tacte S și aplicate intrării S, este nevoie de o variabilă auxiliară de ieșire, pe care o vom nota cu A.

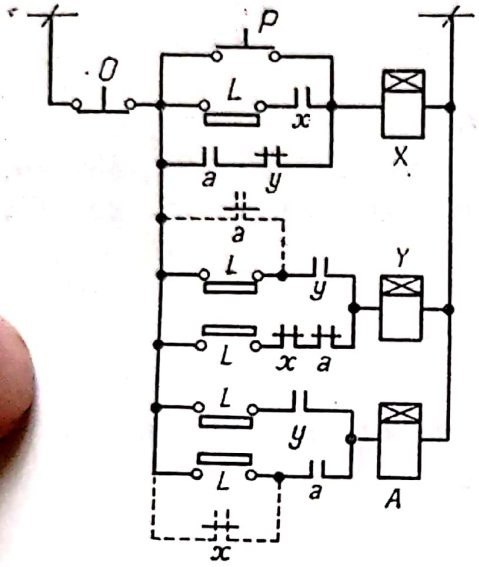


Fig. 11.51. — Schema cu contacte pentru comanda automată a unei acționări electro-hidraulice controlate printr-un singur limitator.

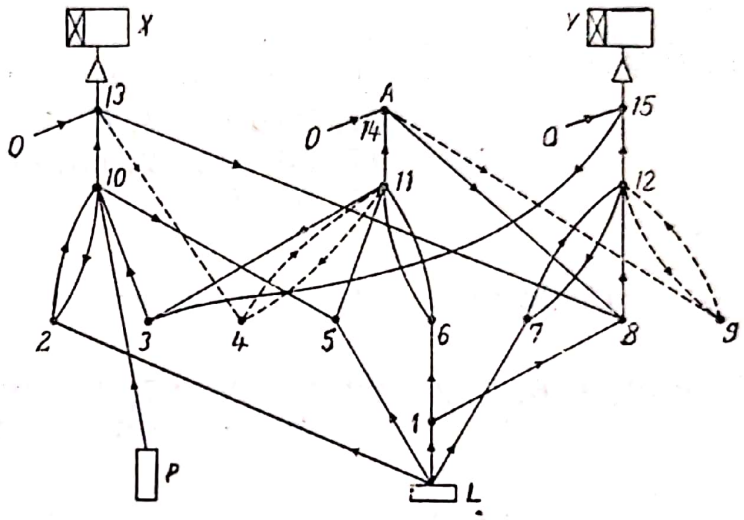


Fig. 11.52. — Schema modulară UNILOG pentru o acționare electrohidraulică.

Avînd în vedere ordinea pornirii electromotoarelor ZYX și cea a opririi lor XYZ, se stabilește diagrama de faze din figura 11.53.

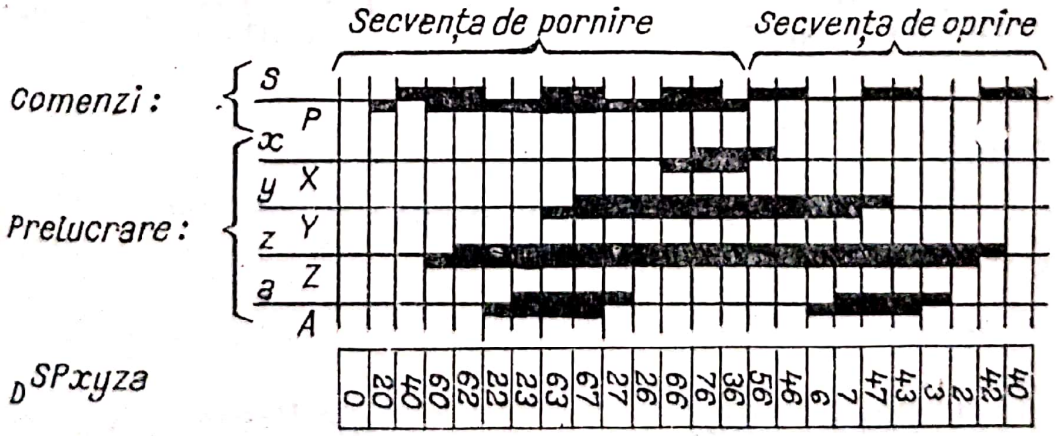


Fig. 11.53. — Diagrama de faze pentru exemplul 11.20.

Controlul tranziţiilor dintre stările schemei de comutaţie sincrone este făcut de către semnalele de tact S.

Codificarea diagramei permite specificarea stărilor de tranziţie şi a celor stabile ale schemei şi, în final, structurarea simbolului de marcare:

$$F(X, Y, Z, A) = D^{SPxyza}_{(36,\ 66,\ 76),\ (6,\ 7,\ 26,\ 27,\ 36,\ 46,\ 56,\ 63,\ 66,\ 67,\ 76),\ (2,\ 3,\ 6,\ 7,\ 22,\ 23,\ 26,\ 27,\ 36,\ 43,\ 46,\ 47,\ 56,\ 60,\ 62,\ 63,\ 66,\ 67,\ 76),\ (6,\ 7,\ 22,\ 23,\ 43,\ 47,\ 63,\ 67)}$$

Stările 0 se extrag din şirul

0, 2, 3, 6, 7, 20, 22, 23, 26, 27, 36, 40, 42, 43, 46, 47, 56, 60, 62, 63, 66, 67, 76.

Expresiile logice minimale vor fi:

— pentru X:

$$4\,\frac{36}{32}\ /\ 76\ /\ 37\ \ 77\ /\ 34\ \ 74\ \ 35\ \ 75, \qquad D^{SPxyza}_{011\underline{11}0} = xP,$$

$$66\ /\ 64\ /\ 76\ \ 74, \qquad D^{SPxyza}_{11\underline{0}1\underline{1}0} = SPy\bar{a};$$

rezultă deci

$$X = Px \cup SPy\bar{a};$$

— pentru Y:

$$2\,\frac{6}{4}\ /\ 7\ /\ 26\ \ 27\ /\ 16\ \ 17\ \ 36\ \ 37, \qquad D^{SPxyza}_{0\underline{0}0\underline{1}1\underline{0}} = \bar{S}y,$$

$$\begin{array}{l} \overset{10}{40\,\underline{46}} \;/\; 56 \;/\; \overset{2}{44}\ \ \overset{20}{54} \;/\; 66\ 76\ 64\ 74, \\ \quad\ \ 6 \end{array} \qquad D\ \overset{SPxyza}{\underline{100110}} = y\bar{a},$$

$$\begin{array}{l} \overset{4}{\underline{63}} \;/\; \overset{2}{67} \;/\; 61\ \overset{10}{65} \;/\; 73\ 77\ 71\ 75, \end{array} \qquad D\ \overset{SPxyza}{110\underline{011}} = SPa,$$

$$\begin{array}{l} \overset{1}{20\,\underline{77}} \;/\; \overset{2}{76} \;/\; 75\ \overset{4}{74} \;/\; 73\ 72\ 71\ 70, \\ 40\,\underline{57} \\ \quad\ \ 37 \\ \quad\ \ 17 \end{array} \qquad D\ \overset{SPxyza}{\underline{111111}} = x;$$

rezultă deci

$$Y = \bar{S}y \cup y\bar{a} \cup SPa \cup (x);$$

— pentru Z:

$$\begin{array}{l} \overset{1}{10\,\underline{2}} \;/\; 3 \;/\; \overset{4}{6}\ \ \overset{20}{7} \;/\; 22\ 23\ 26\ 27, \\ \quad\ 12 \end{array} \qquad D\ \overset{SPxyza}{0\underline{0001}0} = \bar{S}z,$$

$$\begin{array}{l} \overset{2}{20\,\underline{43}} \;/\; 41 \;/\; \overset{4}{47}\ \ \overset{10}{45} \;/\; 53\ 51\ 57\ 55, \\ 40\,\underline{63} \\ \quad\ \ 3 \\ \quad 23 \end{array} \qquad D\ \overset{SPxyza}{\underline{100011}} = a,$$

$$\begin{array}{l} \overset{10}{2\,\underline{46}} \;/\; 56 \;/\; \overset{20}{66}\ \ \overset{1}{76} \;/\; 47\ 57\ 67\ 77, \\ 40\,\underline{44} \\ \quad\ \ 6 \\ \quad\ \ 4 \end{array} \qquad D\ \overset{SPxyza}{\underline{100110}} = y,$$

$$\begin{array}{l} \overset{2}{10\,\underline{60}} \;/\; 62 \;/\; \overset{1}{61}\ \ \overset{4}{63} \;/\; 64\ 66\ 65\ 67, \\ \quad\ 70 \end{array} \qquad D\ \overset{SPxyza}{110\underline{0000}} = SP;$$

rezultă deci

$$Z = \overline{S}z \cup a \cup y \cup SP;$$

— pentru A:

$$\begin{matrix} 1 & 2 & & 10 \\ 6 & 7 & 4 & 5 \end{matrix} \Big/ 16\ 17\ 14\ 15, \qquad D\,\underset{000\underline{1}\underline{1}\underline{0}}{SPxyza} = \overline{S}\overline{P}y,$$

$$\begin{matrix} 1 & & 10 \\ 22 & 23 & 32 & 33 \end{matrix}, \qquad D\,\underset{010\underline{0}\underline{1}\underline{0}}{SPxyza} = \overline{S}P\overline{y}z,$$

$$10\,\frac{\begin{matrix} 4 & & 20 \\ 43 & 47 \end{matrix}}{53} \Big/ 63\ 67 \Big/ 41\ 45\ 61\ 65, \qquad D\,\underset{1\underline{00}\underline{11}}{SPxyza} = Sa,$$

$$2\,\frac{\begin{matrix} 4 & & 20 \\ 33 & 37 \end{matrix}}{31} \Big/ 13\ 17 \Big/ \begin{matrix} & & 40 \\ 73 & 77 & 53 & 57 \end{matrix} \qquad D\,\underset{\underline{01}\underline{10}\underline{11}}{SPxyza} = ax;$$

rezultă deci

$$A = \overline{S}\overline{P}y \cup \overline{S}P\overline{y}z \cup Sa \cup (ax).$$

Schema modulară UNILOG care realizează funcţiunile cerute este dată în figura 11.54.

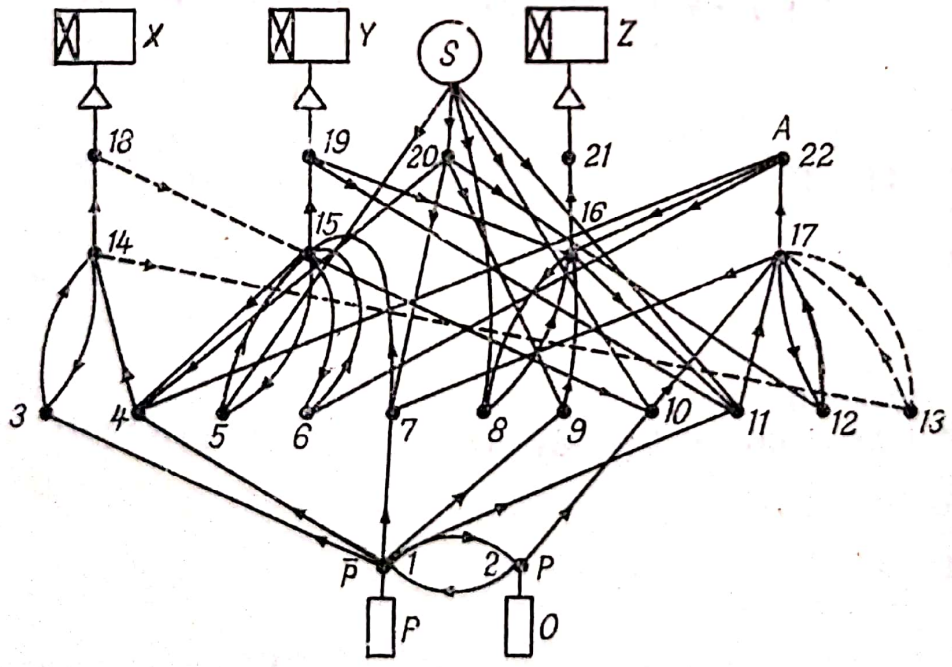


Fig. 11.54. — Schema modulară UNILOG pentru comanda automată sincronă a unui transportor cu bandă.

11.3 SINTEZA UNOR SCHEME DE COMUTAȚIE FOLOSITE ÎN CALCULATOARELE ELECTRONICE ȘI ÎN AUTOMATIZĂRI INDUSTRIALE

Exemplul 11.21. Să se facă sinteza unui circuit de comutație care să dea la ieșire un semnal de aceeași durată cu cel aplict la intrare la fiecare a doua apariție a semnalului de intrare.

În funcționarea acestei scheme se disting patru stări stabile care corespund stărilor ce precedă și succedă aplicarea primului și a celui de-al doilea semnal de intrare. Pentru memorarea acestor patru stări ar fi necesare două variabile secundare, dintre care una ar reprezenta variabila atașată variabilei de ieșire Z și alta ar fi atașată unei variabile secundare auxiliare de ieșire A.

Întrucît însă variabila de ieșire Z trebuie să aibă valoarea binară 1 numai într-una dintre cele patru stări, vor fi necesare două variabile secundare auxiliare de ieșire, și anume A și B.

Notînd prin P variabila primară atașată semnalului de intrare, rezultă următorul simbol de marcare al schemei de comutație:

$$F(A, B, Z) = D^{Pabz} .$$

Ținînd seama de cerințele schemei de comutație, se întocmește diagrama de faze din figura 11.55. Din ea, în urma codificării, rezultă

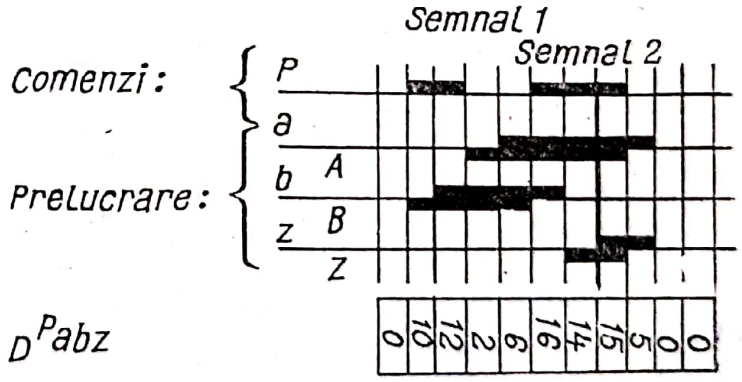


Fig. 11.55. — Diagrama de faze pentru exemplul 11.21.

următorul simbol de marcare:

$$F(A, B, Z) = D^{Pabz}\ (2, 6, 14, 15, 16, 0°, 10°, 5°, 12°),\ (2, 6, 10, 12, 0°, 5°, 14°, 15°, 16°),\ (14, 15, 0°, 2°, 5°, 6°, 10°, 12°, 16°).$$

Explicitat în forma sa minimală, acest simbol ne permite stabilirea următorelor funcţii logice:

— pentru A:

$$\begin{matrix} 4 & 1 \\ 2 & 6 & 3 & 7, \end{matrix} \qquad D \begin{matrix} Pabz \\ 0\underline{0}1\underline{0} \end{matrix} = \overline{P}b,$$

$$\begin{matrix} 1 & 2 \\ 14 & 15 & 16 & 17, \end{matrix} \qquad D \begin{matrix} Pabz \\ 11\underline{0}\underline{0} \end{matrix} = Pa,$$

$$\begin{matrix} 1 & 10 \\ 6 & 7 & 16 & 17, \end{matrix} \qquad D \begin{matrix} Pabz \\ \underline{0}11\underline{0} \end{matrix} = (ab);$$

rezultă deci

$$A = \overline{P}b \cup Pa \cup (ab);$$

— pentru B:

$$\begin{matrix} 4 & 1 \\ 2 & 6 & 3 & 7; \end{matrix} \qquad D \begin{matrix} Pabz \\ 0\underline{0}1\underline{0} \end{matrix} = \overline{P}b,$$

$$\begin{matrix} 2 & 1 \\ 10 & 12 & 11 & 13, \end{matrix} \qquad D \begin{matrix} Pabz \\ 10\underline{0}\underline{0} \end{matrix} = P\bar{a},$$

$$\begin{matrix} 10 & 1 \\ 2 & 12 & 3 & 13, \end{matrix} \qquad D \begin{matrix} Pabz \\ \underline{0}01\underline{0} \end{matrix} = (\bar{a}b);$$

rezultă deci

$$B = \overline{P}b \cup P\bar{a} \cup (\bar{a}b);$$

— pentru Z:

$$\begin{matrix} 1 \\ 14 & 15, \end{matrix} \qquad Z = D \begin{matrix} Pabz \\ 110\underline{0} \end{matrix} = P\bar{b}.$$

Schema echivalentă triunghiulară rezultă din

$$\Delta_A \rightarrow \left| \begin{array}{l} \overset{1}{16} / \overset{2}{17} / 14\ 15 \rightarrow Pa. \\ \overset{4}{2} / \overset{1}{6} / 3\ 7 \rightarrow Pb. \end{array} \right. \quad \Delta_A = Pa \cup Pb,$$

$$\Delta_B \rightarrow \overset{2}{10} / \overset{1}{12} / 11\ 13 \rightarrow P\bar{a}. \quad \Delta_B = P\bar{a},$$

$$\Delta_{AB} \rightarrow \overset{4}{2} / \overset{1}{6} / 3\ 7 \rightarrow \bar{P}. \quad \Delta_{AB} = \bar{P},$$

$$\Delta_{AZ} \rightarrow \overset{1}{14} / \overset{4}{15} / 10\ 11 \rightarrow \bar{b}. \quad \Delta_{AZ} = b.$$

Schema cu contacte triunghiulară este reprezentată în figura 11.56 *a*. Ea prezintă hazard de curse, determinat de contactele lui P. Eliminarea acestui hazard se presupune că se face pin reglarea timpilor de anclanşare a contactelor.

Schema modulară UNILOG este dată în figura 11.56 *b*.

Problema de sinteză de mai sus este rezolvată prin metoda matriceală în [25, p. 73—77].

Exemplul 11. 22. Să se facă sinteza unui dispozitiv indicator de sens de rotaţie care foloseşte drept traductor de sens inelul format din două secţiuni, dat în figura 11.57.

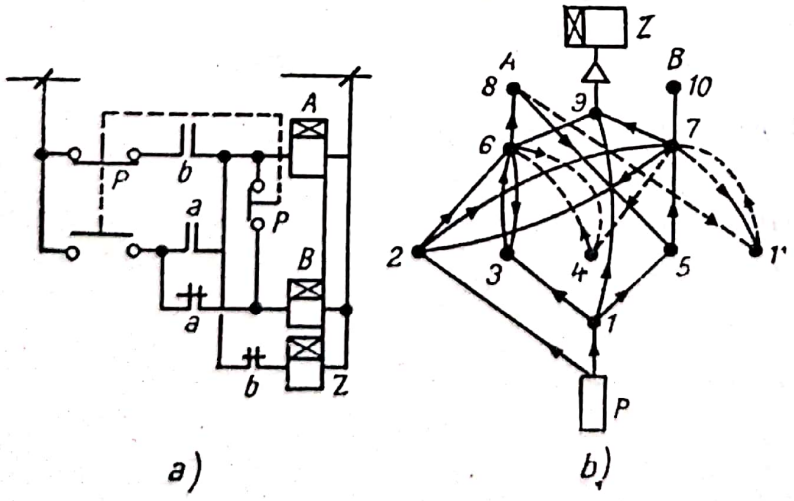


Fig. 11.56. — Circuit de selecţie a unui semnal. a) schema cu contacte b) Schema fără contacte.

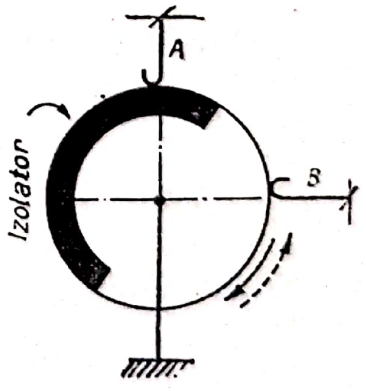


Fig. 11.57. — Traductor de sens de rotaţie.

Din analiza traductorului de sens dat în figura 11.57 rezultă că în cazul rotaţiei orare evoluţia stării variabilelor A şi B este

$$00 \to 10 \to 11 \to 01 \to 00 \to \ldots$$

În cazul rotaţiei antiorare, această evoluţie este dată de

$$00 \to 10 \to 11 \to 01 \to 00 \to \ldots$$

Pentru indicarea sensului de rotaţie se vor folosi două lămpi semnalizatoare, al căror circuit de alimentare este controlat de contactele celor două relee de sens D şi S. Lămpile vor fi stinse numai dacă $A = B = 0$.

Dacă axul se roteşte, din cauza trecerii prin starea (00) una dintre lămpi va pîlpîi. Frecvenţa de pîlpîire este proporţională cu turaţia axului controlat.

Pentru realizarea funcţiilor logice cerute, circuitul de comutaţie va avea drept variabile primare variabilele A şi B, asociate contactelor traductorului de sens şi drept variabile de ieşire şi secundare variabilele asociate celor două relee de sens.

Cu aceste precizări putem forma următorul raport de stare al schemei de comutaţie:

$$\frac{AB\ (ds)}{DS} = \frac{n}{DS}.$$

Stările secvenţei orare ($D = 1$) vor fi date de

$$\frac{0}{00} \to \frac{10}{10} \to \frac{12}{10} \to \frac{16}{10} \to \frac{6}{10} \to \frac{2}{00} \to \frac{0}{00} \to (10) \to \ldots,$$

iar stările secvenţei antiorare ($S = 1$) de

$$\frac{0}{00} \to \frac{4}{01} \to \frac{5}{01} \to \frac{15}{01} \to \frac{11}{01} \to \frac{1}{00} \to \frac{0}{00} \to (4) \to \ldots$$

Scanned with OKEN Scanner

Din analiza stărilor acestor secvențe rezultă următorul simbol de marcare al schemei de comutație:

$$F(D, S) = D\,{}^{ABds}_{D:\,1:\,(6,\,10,\,12,\,16);\;0:\,(0,\,1,\,2,\,4,\,5,\,11,\,15),}$$
$$S:\,1:\,(4,\,5,\,11,\,15);\;0:\,(0,\,1,\,2,\,6,\,10,\,12,\,16).$$

Expresiile booleene minimale deduse din acest simbol vor fi:

— pentru

$$\overset{10}{6} \,/\, \overset{1}{16} \,/\, 7\ 17, \qquad D\,{}^{ABds}_{\underline{0}11\underline{0}} = Bd,$$

$$\overset{2}{10} \,/\, \overset{4}{12} \,/\, 14\ 16, \qquad D\,{}^{ABds}_{1\underline{00}0} = A\bar{s};$$

rezultă deci

$$D = Bd \cup A\bar{s};$$

— pentru $F(S)$:

$$\overset{1}{4} \,/\, \overset{10}{5} \,/\, 14\ 15, \qquad D\,{}^{ABds}_{\underline{0}10\underline{0}} = B\bar{d},$$

$$\overset{2}{11} \,/\, \overset{4}{13} \,/\, 15\ 17, \qquad D\,{}^{ABds}_{1\underline{00}1} = As;$$

rezultă deci

$$S = B\bar{d} \cup As.$$

Schemele cu şi fără contacte sînt date în figura 11.58.

Remarcăm că în cazul schemei modulare UNILOG nu mai este necesară folosirea releelor de sens dacă semnalul de ieşire dat de amplificatoare este suficient de puternic pentru alimentarea lămpilor semnalizatoare.

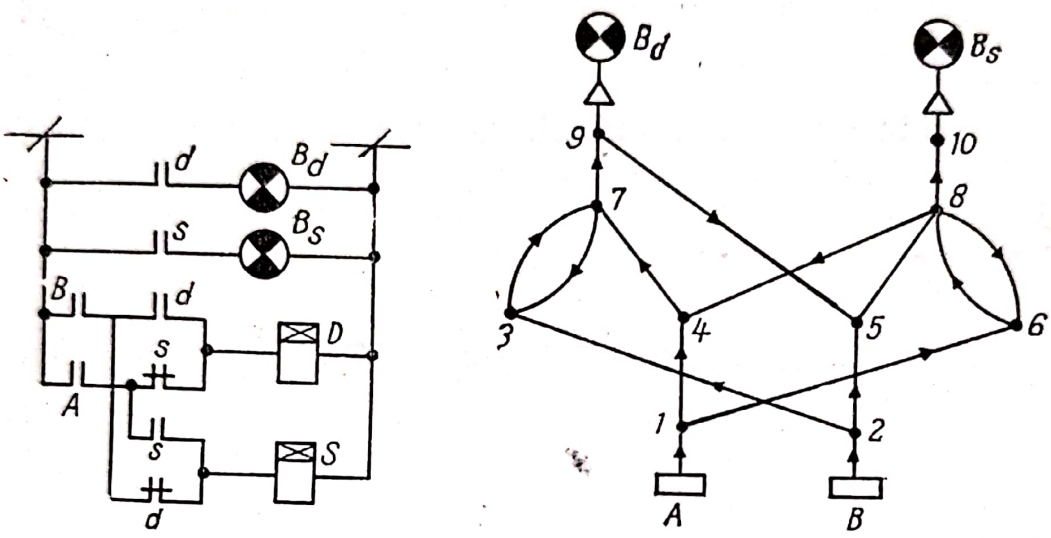


Fig. 11.58. — Indicator de sens de rotaţie.

Exemplul 11.23. Să se facă sinteza unui circuit monoimpulsor.

La intrarea unui astfel de circuit se aplică un tren de impulsuri dreptunghiulare de periadă T.

Prin apăsarea unui buton P, la ieşirea circuitului apare un semnal de aceeaşi formă şi durată cu semnalul din trenul de impulsuri aplicat la intrare.

După apăsarea pe butonul P, momentul apariţiei impulsului de la ieşire coincide cu momentul apariţiei primului impuls întreg din trenul de impulsuri aplicat la intrare.

Cele două situaţii ce se pot ivi, şi anume a) momentul apăsării lui P coincide cu o pauză şi b) momentul apăsării lui P coincide cu un impuls aplicat la intrare, sînt redate de diagrama de faze din figura 11.59.

Variabilele primare A şi P, după cum se vede din această diagramă, pot apărea în patru combinaţii, şi anume 00,01, 10 şi 11. Combinaţia (11) poate să coincidă sau nu cu apariţia semnalului de ieşire S. Pentru a discrimina aceste două situaţii este necesară folosirea a două variabile auxiliare X şi Y. Cu acestea se pot

memora stările care succedă şi precedă apăsarea lui P în cadrul secvenţei 1 şi situaţiile similare din cadrul secvenţei 2.

Simbolul de marcare al schemei va rezulta din considerarea stărilor apărute prin codificarea diagramei de faze din figura 11.59. El este

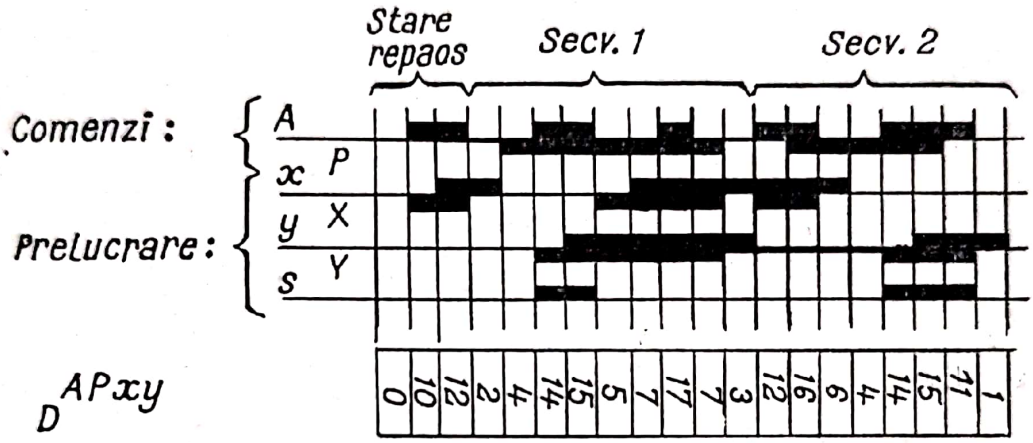


Fig. 11.59 — Diagrama de faze pentru exemplul 11.23.

$$F(X, Y, S) = D^{APxy}_{\substack{1:(5,7,10,12,16,17);\ 0:(0,1,3,4,6,11,14,15),\\ 1:(5,7,11,14,15,17);\ 0:(0,1,3,4,6,10,12,16),\\ 1:(11,14,15);\ 0:(0,1,3,4,5,6,7,10,12,16,17).}}$$

Rezultă următoarele expresii logice minimale:

— pentru X:

$$\overset{2}{5\,/\,7} \qquad D^{APxy}_{0\,1\,\underline{0}\,1} = \overline{A}Py,$$

$$\overset{2}{10\,/\,12} \qquad D^{APxy}_{1\,0\,\underline{0}\,0} = AP\overline{y},$$

$$\overset{1\quad 4}{16\,/\,17\,/\,12\,13} \qquad D^{APxy}_{\underline{1}\,1\,1\,\underline{0}} = Ax,$$

$$\overset{10}{7\,/\,17} \qquad D^{APxy}_{\underline{0}\,1\,1\,1} = (Pxy);$$

rezultă deci

$$X = \overline{A}Py \cup AP\overline{y} \cup Ax \cup (Pxy);$$

— pentru Y:

$$\begin{matrix} 2 & 10 \\ 5 & 7 & 15 & 17 \end{matrix} \qquad D^{APxy}_{\underline{0}1\underline{0}1} = Py,$$

$$\begin{matrix} 2 & 4 \\ 11 & 13 & 15 & 17 \end{matrix} \qquad D^{APxy}_{1\underline{00}1} = Ay,$$

$$\begin{matrix} 1 \\ 14 & 15 \end{matrix} \qquad D^{APxy}_{110\underline{0}} = AP\bar{x};$$

rezultă deci

$$Y = y(P \cup A) \cup AP\bar{x}.$$

— pentru S:

$$\begin{matrix} 4 \\ 11 & 15 \end{matrix} \qquad D^{APxy}_{1\underline{0}01} = A\bar{x}y,$$

$$\begin{matrix} 1 \\ 14 & 15 \end{matrix} \qquad D^{APxy}_{110\underline{0}} = AP\bar{x};$$

— rezultă deci

$$S = A\bar{x}(P \cup y).$$

Schemele de comutaţie de tipul sistetizat în acest exemplu se realizează, de regulă, numai cu elemente de comutaţie statică. Pe baza expresiilor logice obţinute s-a structurat schema din figura 11.60.

Problema sintezei acestei scheme prin metoda matriceală a lui Huffman [47] este dată în lucrarea [72, p. 315].

Exemplul 11.24. Să se facă sinteza blocului sumator-scăzător elementar (pentru doi biți) dintr-un calculator electronic numeric.

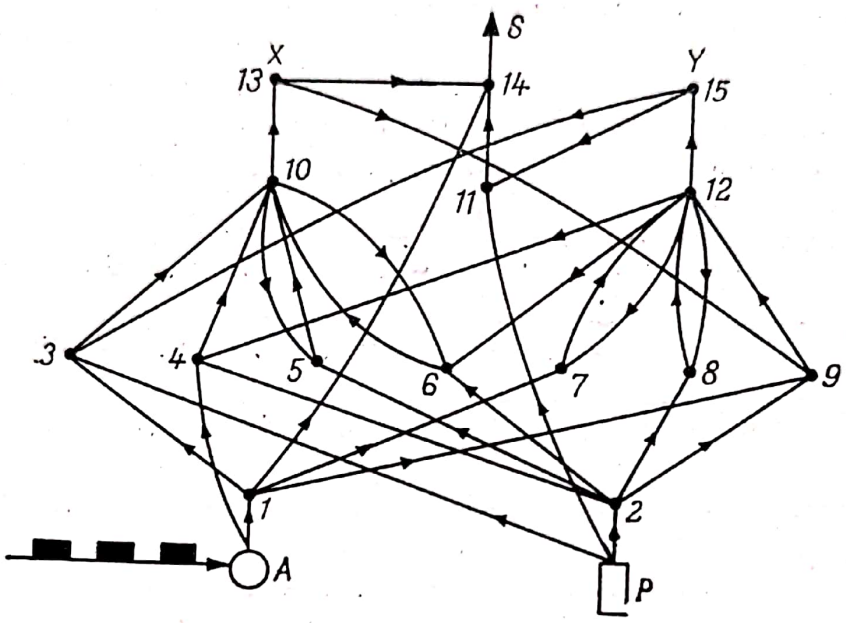


Fig. 11.60. — Schema unui circuit monoimpulsor.

Comanda de sumare sau de scădere se dă printr-un semnal D. Pentru $D = 0$, blocul efectuează sumarea biților aplicați la intrare, iar pentru $D = 1$, scăderea acestora.

Tabelul 11.1

n_k	D	a	b	t	R	T
0	0	0	0	0	0	0
1	0	0	0	1	1	0
2	0	0	1	0	1	0
3	0	0	1	1	0	1
4	0	1	0	0	1	0
5	0	1	0	1	0	1
6	0	1	1	0	0	1
7	0	1	1	1	1	1
10	1	0	0	0	0	0
11	1	0	0	1	1	1
12	1	0	1	0	1	1
13	1	0	1	1	0	1
14	1	1	0	0	1	0
15	1	1	0	1	0	0
16	1	1	1	0	0	0
17	1	1	1	1	1	1

Scanned with OKEN Scanner

Pe baza tabelei de adevăr 11.1 se stabileşte următorul simbol de marcare al blocului:

$$F(R, T) = D^{Dabt}_{(1, 2, 4, 7, 11, 12, 14, 17),\ (3,5, 6, 7, 11, 12, 13, 17).}$$

Alegînd calea sintezei prin grupe de numere de stare adiacente care corespund valorii binare 0 a variabilei de ieşire, se ajunge la următoarele expresii logice minimale:

$$R = \overline{\bar{a}\bar{b}\bar{t} \cup \bar{a}bt \cup a\bar{b}t \cup ab\bar{t}} = \overline{\bar{a}\ (b \oplus t)} \top a\,(b \oplus t) = a \oplus (b \oplus t),$$

$$T = \overline{\bar{b}\bar{t} \cup \bar{D}\bar{a}\bar{b}\,(t) \cup \bar{D}\bar{a}\,(b)\,\bar{t} \cup Da\bar{b}\,(t) \cup Da\,(b)\,\bar{t}} =$$

$$= \overline{\bar{b}\bar{t} \cup (Da \cup \bar{D}\bar{a})\,(b \oplus t)} = \bar{b}\bar{t} \top \overline{(D \oplus a)}\,(b \oplus t).$$

Schema modulară UNILOG a blocului este dată în figura 11.61. Constatăm că pentru realizarea funcţiei logice „sau exclusiv" negate sînt necesare patru module „nici".

La ieşirea modulului 8 se obţine

$$R = \overline{\bar{a}\ (b \oplus t)} \cdot \overline{a\ (b \oplus t)}\ [a \cup (b \oplus t)]\ [\bar{a} \cup \overline{(b \oplus t)}] =$$

$$= \bar{a}\,(b \oplus t) \cup a\,\overline{(b \oplus t)} = a \oplus (b \oplus t).$$

Rezultatul R corespunde sumei sau diferenţei modulo 2 a biţilor a şi b, precum şi transferului t, aplicat la intrarea blocului.

La ieşirea modulelor 9 şi 10 se obţin

$$\bar{D} \cdot \overline{\bar{a}\ (b \oplus t)} \qquad \text{şi} \qquad D \cdot \overline{a\,(b \oplus t)},$$

astfel încît la ieşirea modulului 12 avem

$$\overline{\bar{D} \cdot \overline{\bar{a}\ (b \oplus t)}} \cdot \overline{D \cdot \overline{a\,(b \oplus t)}} = [D \cup \bar{a}\,(b \oplus t)]\ [D \cup a\,(b \oplus t)] =$$

$$= \bar{a}\,\bar{D}\,(b \oplus t) \cup aD\,(b \oplus t) = \overline{(a \oplus \bar{D})}\,(b \oplus t).$$

Transferul T apare astfel la ieşirea modulului 13. El corespunde sumei pentru cazul cînd $D = 0$ sau diferenţei pentru cazul cînd $D = 1$.

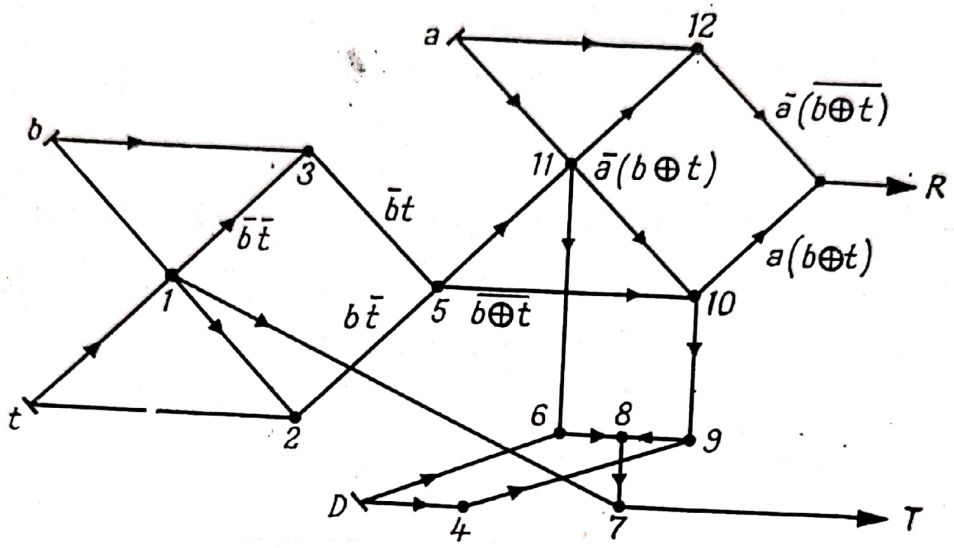


Fig. 11.61. — Bloc sumator-scăzător în execuţie modulară UNILOG. D-Comandă pentru scăzător.

Exemplul 11.25. Să se facă sinteza unui element al blocului de comandă a unui element de memorie dintr-un calculator electronic numeric (fig. 11.62). Specificaţiile schemei sînt:

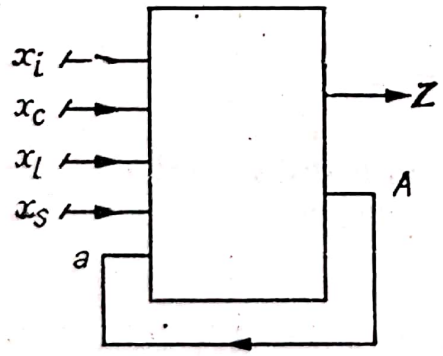


Fig. 11.62. — Element de comandă.

a) Dacă există numai una dintre mărimile de intrare: x_i, x_c, x_l sau x_s, starea schemei nu se modifică.

b) Dacă există simultan semnalul de înregistrat x_i şi cel de comandă a înregistrării x_c, memoria A se înscrie şi rămîne înscrisă chiar după ce $x_i = x_c = 0$.

c) Memoria A fiind înscrisă, dacă apare semnalul de lectură x_l, blocul furnizează la ieşirea Z informaţia memorată.

d) Memoria A fiind înscrisă, dacă apare semnalul de ştergere a memoriei x_s, memoria se şterge.

Avînd în vedere variabilele de intrare şi cele de ieşire, raportul de stare va fi

$$\frac{x_i\, x_c \cdot x_l x_s\, a}{ZA}$$

În formă codificată, specificaţiile schemei se scriu astfel

Specificaţia a)

$$\frac{0}{0};\ \frac{20}{0};\ \frac{10}{0};\ \frac{4}{0};\ \frac{2}{0}$$

Specificaţia b)

$$0 \xrightarrow{x_i=x_c=1} \frac{30}{1} \xrightarrow{a=1} \frac{31}{1} \xrightarrow{x_i=x_c=0} \frac{1}{1}$$

Specificaţia c)

$$1 \xrightarrow{x_l=1} \frac{5}{3}$$

Specificaţia d)

$$1 \xrightarrow{x_s=1} \frac{3}{0} \xrightarrow{a=0} 2$$

Datele culese din stările codificate date mai sus ne permit structurarea următorului simbol de marcare:

$$F(Z, A) = D\,\frac{x_i\, x_c \cdot x_1\, x_s\, a}{\begin{array}{l} Z : 1 : (5)\ 0 : (0,\ 1,\ 2,\ 3,\ 4,\ 10,\ 20,\ 30,\ 31) \\ A : 1 : (1,\ 5,\ 30,\ 31),\ 0 : (0,\ 2,\ 3,\ 4,\ 10,\ 20) \end{array}}$$

Sinteza prin grupe de 1, ne conduce la grupările:

Pentru Z

$$20\,\frac{5}{25}\,/\,\overset{2}{7}\,/\,\overset{10}{15}\ 17 \longrightarrow \underline{00}.1\underline{01} \rightarrow x_l\, a.$$

Pentru A

$$20\,\frac{1}{21}\,/\,5\,/\,\overset{4}{11}\;\overset{10}{15} \longrightarrow \underline{00}.\underline{0}01 \to \bar{x}_s\, a.$$

$$4\,\frac{30}{34}\,/\,\overset{1}{2}\,/\,\overset{31}{32}\;33 \longrightarrow 11.\underline{000} \to x_i\, x_c.$$

Rezultă deci funcţiile logice

$$Z = x_l\, a = \bar{x}_1 \top \bar{a}.$$

$$A = \bar{x}_S\, a \cup x_i\, x_c = \overline{(x_s \top \bar{a}) \top (\bar{x}_i \top \bar{x}_c)}.$$

Sinteza prin grupe de 0, ne conduce la grupările:

Pentru Z

$$20\,\frac{0}{20}\,/\,\overset{1}{1}\,/\,\overset{2}{2}\;\overset{10}{3}\,/\,10\;11\;12\;13 \longrightarrow \underline{00}.\underline{000} \to \bar{x}_1.$$

Pentru A și Z

$$10\,\frac{0}{10}\,/\,\overset{2}{2}\,/\,\overset{4}{4}\;6 \longrightarrow \underline{00}.\underline{000} \to \bar{x}_i\,\bar{a}.$$

Pentru A

$$20\,\frac{3}{23}\,/\,\overset{1}{2}\,/\,\overset{4}{7}\;\overset{10}{6}\,/\,13\;12\;17\;16 \longrightarrow \underline{00}.\underline{011} \to x_s.$$

$$20\,/\,\overset{1}{21}\,/\,\overset{2}{22}\;\overset{4}{23}\,/\,24\;25\;26\;27 \longrightarrow 10.\underline{000} \to \bar{x}_s\, x_i.$$

Rezultă expresiile logice:

$$Z = \overline{\bar{x}_l \top \bar{x}_i \bar{a}} = \bar{x}_l \top (x_i \top a).$$

$$A = \overline{\bar{x}_i \bar{a} \cup x_s \cup \bar{x}_c x_i} = (x_i \top a) \top x_s \top (x_c \top \bar{x}_i).$$

Primul rezultat obţinut corespunde unui raport de structură a schemei UNILOG dat de:

$$\frac{8 \text{ module}}{12 \text{ intrări}}$$

în timp ce al doilea rezultat, dă un raport de structură de:

$$\frac{6 \text{ module}}{11 \text{ intrări}}$$

deci mai avantajos din punct de vedere economic.

Schema modulară UNILOG, ce corespunde celei de a doua soluţii este dată în figura 11.63.*a* iar cea cu elemente NU, SI, SAU, dedusă prin folosirea primei soluţii, este dată în figura 11.63.b.

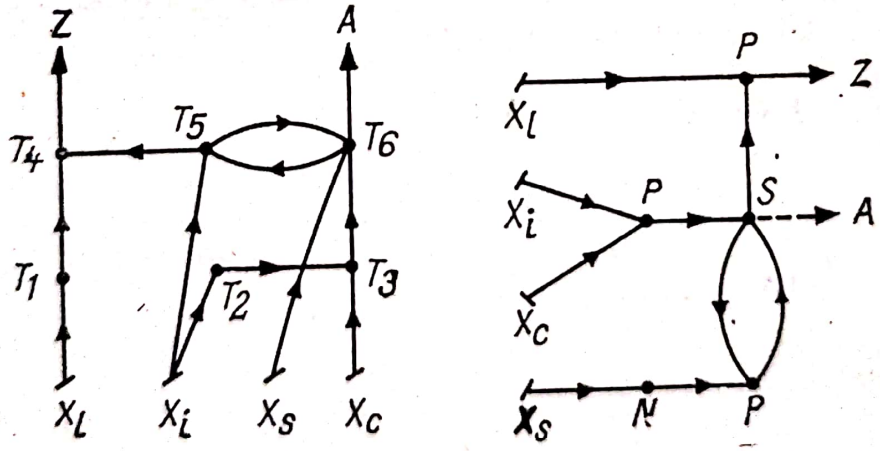


Fig. 11.63. — Schema modulară pentru exemplul 11.25. a) UNILOG, b) cu elemente *PSN*.

În schema a doua notaţiile nodurilor corespund respectiv funcţiilor logice efectuate de acestea *P*-produs, *S*-sumă şi *N*-negaţie.

Exemplul 11.26. Se cere găsirea structurii unui circuit secvențial sincron cu patru borne de intrare: $x_1x_2x_3x_4$ și o bornă de ieșire Z, la care apare semnal numai dacă semnalele de intrare s-au aplicat în ordinea: $x_1x_2x_3$.

Semnalele de intrare nu se aplică niciodată simultan. Ele se aplică sincron cu semnalele de tact.

Se impune întîi stabilirea numărului stărilor stabile ale circuitului.

Potrivit specificației din enunț se distinge starea inițială s_0, starea s_1, care succede aplicării variabilei x_1, starea s_2, care succede aplicării variabilei x_2 și starea finală s_3, care succede aplicării variabilei x_3. Pentru discriminarea celor 4 stări stabile ale circuitului sînt necesare două variabile secundare.

În cazul circuitelor sincrone, variabilele secundare se obțin prin intermediul unor memorii. Înscrierea sau ștergerea acestor memorii se face în ritmul semnalelor de tact care controlează evoluția sincronă a schemei.

Presupunem că în cazul nostru, pentru obținerea celor două variabile secundare y_1 și y_2 se folosesc două memorii (circuite basculante bistabile) avînd intrările de înscriere I_1 și I_2 și cele de ștergere S_1 și S_2.

Cu aceste precizări putem structura următorul raport de stare al circuitului:

$$\frac{x_1x_2x_3 \cdot x_4y_1y_2}{I_1S_1 \cdot I_2S_2 \cdot Z}.$$

Plecînd de la acest raport de stare, procedăm la codificarea specificațiilor schemei.

Specificația 1. Presupunem că starea inițială s_0, se caracterizează prin $y_1 = y_2 = 0$. Această stare se obține prin ștergerea celor două memorii cu ajutorul unui semnal exterior de „zero". În această stare este obligator ca: $I_1 = I_2 = 0$, valorile binare ale lui S_1 și S_2 fiind indiferente. În starea s_0, avem $Z = 0$.

Codificat, aceste informații sînt cuprinse în formă simbolică astfel:

$$\frac{0}{0x \cdot 0x \cdot 0}$$

Specificația 2. Trecerea circuitului din starea s_0 în starea s_1, caracterizată prin : $y_1=0$, $y_2 = 1$, este posibilă numai dacă $x_1 = 1$. Dacă $x_2 = 1$ sau $x_3 = 1$, sau $x_4 = 1$, starea circuitului nu se modifică.

Întrucît pentru a înscrie memoria Y_2, este necesar ca $I_2 = 1$ și $S_2 = 0$, vom putea scrie

$$\frac{40}{0x.10.0}; \frac{20}{0x.0x.0}; \frac{10}{0x.0x.0}; \frac{4}{0x.0x.0}$$

Specificația 3. Trecerea circuitului din starea s_1 în starea s_2, caracterizată prin $y_1=y_2 = 1$, este posibilă numai dacă $x_2 = 1$. Aceasta se exprimă codificat astfel

$$\frac{21}{10.x0.0}$$

După ce $y_2 = 1$, valoarea binară a lui I_2 este indiferentă, întrucît memoria a fost înscrisă. În schimb însă, obligator este necesar ca $S_2 = 0$.

Dacă în starea s_1, apare $x_1 = 1$, starea circuitului nu se modifică, iar dacă apare $x_3 = 1$ sau $x_4 = 1$, circuitul trece în starea s_0. Codificat aceasta se exprimă astfel

$$\frac{41}{0x.x0.0}; \frac{11}{0x.01.0}; \frac{5}{0x.01.0}.$$

Specificația 4. Trecerea circuitului din starea s_2 în starea finală s_3, caracterizată prin : $y_1 = 1$, $y_2 = 0$ și $Z = 1$, este posibilă numai dacă $x_3 = 1$.

Dacă în starea s_2, apare $x_1 = 1$, circuitul trece în starea s_1 iar dacă apare $x_2 = 1$ sau $x_4 = 1$, el revine în starea s_0. Codificat, aceasta se exprimă astfel

$$\frac{43}{01.x0.0}; \frac{23}{01.01.0}; \frac{13}{x0.01.0}; \frac{4}{01.01.0}.$$

Specificația 5. Dacă în starea s_3 apare $x_1 = 1$ sau $x_2 = 1$ sau $x_3 = 1$ sau $x_4 = 1$, starea circuitului nu se mai modifică și $Z = 1$. El poate fi readus în starea s_0 numai printr-un semnal aplicat pe intrarea „zero".

Codificat, aceasta se exprimă astfel

$$\frac{42}{x0.0x.1}; \frac{22}{x0.0x.1}; \frac{12}{x0.0x.1}; \frac{6}{x0.0x.1}.$$

Folosind informațiile cuprinse în stările codificate mai sus, stabilim următorul simbol de marcare al circuitului

$$F(I_1, S_1, I_2, S_2, Z) = D \begin{matrix} x_1x_2x_3.x_4y_1y_2 \\ I_1 : 1(21)\,0 : (0, 4, 5, 7, 10, 11, 20, 23, 40, 41, 43) \\ S_1 : 1 : (7, 23, 43)\,0 : (6, 12, 13, 21, 22, 42) \\ I_2 : 1 : (40)\,0 : (0, 4, 5, 6, 7, 10, 11, 12, 13, 20, 22, 23, 42) \\ S_2 : 1 : (5, 7, 11, 13, 23)\,0 : (21, 40, 41, 43) \\ Z : 1 : (6, 12, 22, 42)\,0 : 4, 5, 7, 10, 11, 13, 20, 21, 23, 40, 41, 43) \end{matrix}$$

Pentru cele 5 funcții logice ale schemei se pot forma următoarele grupări de numere de stare adiacente, care corespund valorilor binare 1 ale variabilelor de ieșire:

Pentru I_1

$$\begin{matrix} & 4 & 10 & & 40 & \\ 21 & / & 25 & / & 31\ 35 & / & 61\ 65\ 71\ 75 \end{matrix} \longrightarrow \underline{0}10.\underline{0}01 \rightarrow x_2\overline{y}_1y_2.$$

Pentru S_1

$$\begin{matrix} & 4 & 20 & & 40 & \\ 7 & / & 3 & / & 27\ 23 & / & 47\ 43\ 67\ 63 \end{matrix} \longrightarrow \underline{00}0.\underline{1}11 \rightarrow \overline{x}_3y_1y_2.$$

Pentru I_2

$$\begin{matrix} & & 1 & & 4 & & 10 & \\ 20 & \underline{40} & / & 41 & / & 44\ 45 & / & 50\ 51\ 54\ 55 \\ & \underline{60} & \multicolumn{5}{l}{\ldots\ldots\ldots\ldots\ldots\ldots} \end{matrix} \longrightarrow 1\underline{00}.\underline{00}0 \rightarrow x_1\overline{y}_1.$$

Pentru S_2

$$1\,\frac{5}{4}\,\Big/\,\overset{2}{7}\,\Big/\,\overset{4}{13}\,\Big/\,\overset{10}{15\ 17\ 11\ 13} \longrightarrow 000.\underline{101} \to \bar{x}_1\bar{x}_2.$$

$$20\,\frac{23}{2}\,\Big/\,\overset{1}{22}\,\Big/\,\overset{4}{27\ 26}\,\Big/\,\overset{10}{33\ 32\ 37\ 36} \longrightarrow \underline{010}.\underline{011} \to \bar{x}_1 y_1.$$

Pentru Z

$$40\,\frac{6}{46}\,\Big/\,\overset{4}{2}\,\Big/\,\overset{10}{16\ 12}\,\Big/\,\overset{20}{26\ 22\ 36\ 32} \longrightarrow \underline{000}.\underline{110} \to y_1\bar{y}_2.$$

Rezultă deci următoarele funcţii logice

$$I_1 = x_2\bar{y}_1 y_2. \qquad I_2 = x_1\bar{y}_1.$$

$$Z = y_1 y_2.$$

$$S_1 = \bar{x}_3 y_1 y_2. \qquad S_2 = \bar{x}_1(\bar{x}_2 \cup \bar{y}_1)$$

Observaţie. Pînă acum, de regulă, ordinea luării în considerare a diferenţelor de adiacenţă pentru formarea grupelor de numere de stare adiacente era cea progresiv crescătoare. Aşa s-a procedat şi la rezolvarea problemei de mai sus.

Se poate totuşi stabili şi o altă ordine de considerare a diferenţelor de adiacenţă în funcţie de scopul urmărit. În acest sens, se calculează în prealabil numărul total de biţi 1 ce revin numerelor de stare permise respectiv celor interzise pentru valoarea 1 a variabilei de ieşire.

Spre exemplu, pentru S_2 găsim pentru numerele de stare permise :

5 biţi 1 ce corespund diferenţei de adiacenţă 1,
3 „ 2,
2 „ 4,
2 „ 10,
1 bit 1 ce corespunde diferenţei de adiacenţă 20.

Pentru numerele de stare interzise găsim :

3 biţi 1 ce corespund diferenţei de adiacenţă 1,
3 ,, 40,
1 bit 1 ce corespunde diferenţei de adiacenţă 2,
1 bit 1 care corespunde diferenţei de adiacenţă 20.

Dacă folosim diferenţele de adiacenţă în ordinea descrescătoare a numărului de biţi 1 conţinuţi de numerele de stare permise, atunci numerele de stare permise se vor cuprinde într-un număr minim de grupări.

Dacă folosim diferenţele de adiacenţă în ordinea descrescătoare a numărului de biţi 1 a numerelor de stare interzise, numerele de stare permise se vor cuprinde în grupe avînd un număr maxim de numere de stare.

Cele două cazuri corespund respectiv scopului obţinerii unui număr minim de implicanţi primi sau scopului obţinerii unui număr minim de variabile continuate de implicanţii primi ai expresiei logice minimale.

Pentru S_2, se obţine în primul caz formula

$$S_2 = \bar{x}_1\bar{x}_2 \cup \bar{x}_1 y_1.$$

În cazul al doilea, dînd prioritate folosirii diferenţelor de adiacenţă în ordinea : 1, 40, 2, 20,... se obţin grupările

```
      1  40      2
      5 / 4 / 45  44 / 7  6  47  46
20   25 . . . . . . . . . . . . . .   → 000.110 → x4.
10   15 . . . . . . . . . . . . . .
     35 . . . . . . . . . . . . . .

      1  40      2
     11 / 10 / 51  50 / 13 12 53 52
20   31 . . . . . . . . . . . . . .   → 010.011 → x2y1.
 4   15 . . . . . . . . . . . . . .
     35 . . . . . . . . . . . . . .
```

Se obţine pe această cale formula minimală

$$S_2 = x_2 y_1 \cup x_3 \cup x_4,$$

formulă care este dată şi în lucrarea [91].

Schema de comutaţie statică, care corespunde expresiilor logice obţinute, este dată în figura 11.64.

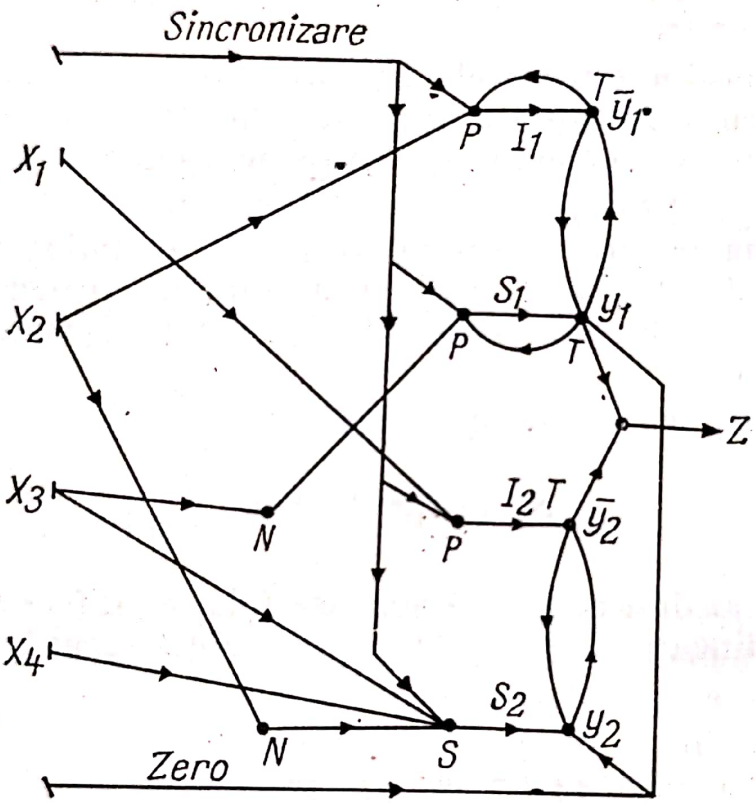


Fig. 11.64. — Schema aferentă exemplului 11.27.

Notaţiile nodurilor corespund respectiv cu operaţiile logice *P*-produs, *S*-sumă şi *N*-negaţie, *T*-operaţie NICI.

Funcţionarea sincronă a schemei este asigurată prin impulsurile de tact ce vin prin conductorul „sincronizare". Aceste impulsuri deschid porţile *SI* prin care se aplică semnalele de înscriere sau de ştergere a celor două memorii.

Exemplul 11.28. Ne propunem să facem sinteza numărătorului în cod binar.

Dacă la intrarea unui circuit de comutaţie se aplică un semnal x avînd forma de impulsuri dreptunghiulare separate prin pauze, la ieşirile acestuia $Y_1, Y_2, \ldots Y_n$, se memorează numărul binar egal cu numărul impulsurilor aplicate la intrare.

Se cere să găsim expresiile logice ale variabilelor de înscriere a celor n memorii formate din circuite basculante bistabile cu o singură intrare.

Se are în vedere faptul că bascularea circuitelor poate avea loc numai în momentul cînd semnalul aplicat la intrarea unică a memoriei variază de la 1 la 0.

Considerăm mai întîi cazul cînd $n = 3$.

Avînd în vedere datele enunţului problemei de sinteză, structurăm următorul raport de stare al schemei de comutaţie

$$\frac{x \cdot y_1 y_2 y_3}{Y_1 Y_2 Y_3}.$$

Dacă în starea 0/0, apare $x = 1$, se trece în starea 10/1 prin care este pregătită bascularea memoriei Y_1, basculare ce are loc atunci cînd $x = 0$ şi se trece în starea 1/2, prin care este pregătită bascularea memoriei Y_2.

Codificat, această secvenţă de lucru se scrie astfel

$$\frac{0}{0} \xrightarrow{x=1} \frac{10}{1} \xrightarrow{x=0} \frac{1}{2}.$$

Dacă în starea 1/2, apare $x = 1$, se trece în starea 11/3, prin care sînt pregătite pentru basculare memoriile Y_1 şi Y_2, fapt ce are loc atunci cînd $x = 0$ şi se trece în starea 2/4, prin care este pregătită bascularea memoriei Y_3.

Codificat, stările secvenţei se succed astfel

$$1 \xrightarrow{x=1} \frac{11}{3} \xrightarrow{x=0} \frac{2}{4}.$$

Scanned with OKEN Scanner

Raţionamente similare ne conduc la stabilirea următoarelor stări ale secvenţei de lucru, date de :

$$2 \xrightarrow{x=1} \frac{12}{5} \xrightarrow{x=0} \frac{3}{6} \xrightarrow{x=1} \frac{13}{7} \xrightarrow{x=0} \frac{4}{0} \xrightarrow{x=1} \frac{14}{1} \xrightarrow{x=0} \frac{5}{2} \xrightarrow{x=1} \frac{15}{3}$$

$$\xrightarrow{x=0} \frac{6}{4} \xrightarrow{x=1} \frac{16}{5} \xrightarrow{x=0} \frac{7}{6} \xrightarrow{x=1} \frac{17}{7} \xrightarrow{x=0} 0.$$

Analiza stărilor acestei secvenţe ne permite structurarea următorului simbol de marcare

$$F(Y_1, Y_2, Y_3) = D^{x \cdot y_1 y_2 y_3}_{(2, 3, 6, 7, 12, 13, 16, 17), (1, 3, 5, 7, 11, 13, 15, 17), (10, 11, 12, 13, 14, 15, 16, 17).}$$

Funcţiile logice ale celor trei variabile de înscriere a memoriilor, rezultă din grupările :

Pentru Y_1

$$\begin{array}{l} \ 1\ \ 4\ \ \ 10 \\ 2\ /\ 3\ /\ 6\ \ 7\ /\ 12\ \ 13\ \ 16\ \ 17 \end{array} \qquad \rightarrow \underline{0.010} \rightarrow y_2.$$

Pentru Y_2

$$\begin{array}{l} \ 2\ \ 4\ \ \ 10 \\ 1\ /\ 3\ /\ 5\ \ 7\ /\ 11\ \ 13\ \ 15\ \ 17 \end{array} \qquad \rightarrow \underline{0.001} \rightarrow y_1.$$

Pentru Y_3

$$\begin{array}{l} \ 1\ \ \ 2\ \ \ \ \ 4 \\ 10\ /\ 11\ /\ 12\ \ 13\ /\ 14\ \ 15\ \ 16\ \ 17 \end{array} \rightarrow \underline{1.000} \rightarrow x.$$

Rezultă deci în general

$$Y_k = y_{k-1} \qquad \text{şi} \qquad Y_n = x.$$
$$(k = 1,2, \ldots, n-1)$$

Aceste formule permit structurarea schemei numărătorului binar, dată în figura 11.65.

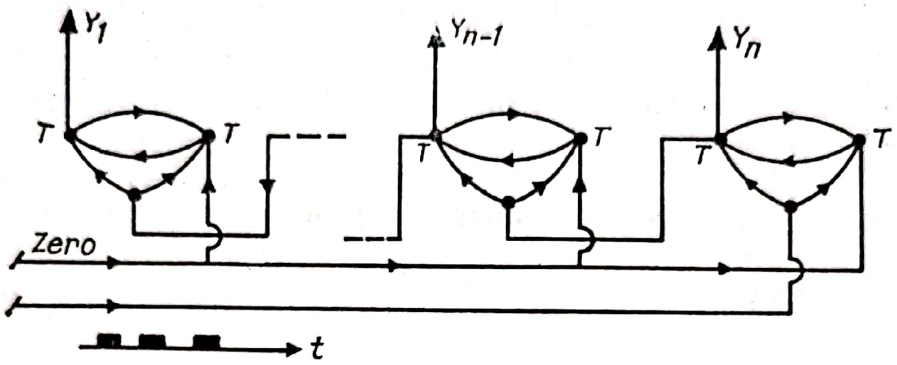


Fig. 11.65. — Schema numărătorului binar.

Exemplul 11.29. Ne propunem să facem sinteza unui numărător de 7 în cod Gray, care revine în starea inițială după 7 impulsuri aplicate la intrare.

Codul Gray este acel cod binar la care trecerea de la un număr la numărul următor nu modifică decît valoarea binară a unui singur bit al codului. Acest cod este folosit mai ales la convertoarele analog-numerice, întrucît elimină pericolul apariției fenomenelor de hazard la comutarea simultană a valorilor binare a mai multor variabile.

Pentru a stabili raportul de stare este necesar să facem bilanțul stărilor necesare ale acestuia. Astfel, pentru discriminarea unui impuls din șirul de impulsuri sînt necesare trei stări : o stare anterioară aplicării impulsului, o stare ce succede aplicării acestuia și o stare pe durata existenței acestui impuls. Întrucît starea ce succede aplicării unui impuls coincide cu starea ce precede impulsul următor, pentru N impulsuri, vor fi necesare $2N+1$ stări.

În cazul nostru, $N = 7$ și vor fi deci necesare 4 memorii pentru memorarea celor $2.7 + 1 = 15$ stări. Rezultă deci următorul raport de stare :

$$\frac{xy_1 \cdot y_2y_3y_4}{I_1S_1 \cdot I_2S_2 \cdot I_3S_3 \cdot I_4S_4}$$

unde, prin I_k, s-au notat variabilele de înscriere a memoriilor Y_k iar prin S_k, cele de ștergere a lor.

Alegem următorul cod cu 4 biţi reprezentativi :

0, 4, 14, 10, 11, 15, 5, 1, 3, 7, 17, 13, 12, 16, 6, 2.

Plecînd de la raportul de stare şi avînd în vedere codul progresiv adoptat (la care se observă că numerele se succed în şir în aşa fel încît diferă între ele doar prin valoarea unui bit) se stabileşte uşor următoarea secvenţă de lucru a numărătorului :

Primul impuls

$$\frac{0}{0x.0x.0x.0x} \xrightarrow{x=1} \frac{20}{0x.10.0x.0x} \xrightarrow{v_2=1} \frac{24}{0x.x0.0x.0x} \xrightarrow{x=1}$$

$$\frac{4}{10.x0.0x.0x} \xrightarrow{v_1=1} \frac{14}{x0.x0.0x.0x}.$$

Al doilea impuls

$$14 \xrightarrow{x=1} \frac{34}{x0.01.0x.0x} \xrightarrow{v_2=1} \frac{30}{x0.0x.0x.0x} \xrightarrow{x=0} \frac{10}{x0.0x.0x.10}$$

$$\xrightarrow{v_4=1} \frac{11}{x0.0x.0x.x0}.$$

Al treilea impuls

$$11 \xrightarrow{x=1} \frac{31}{x0.10.0x.x0} \xrightarrow{v_2=1} \frac{35}{x0.x0.0x.x0} \xrightarrow{x=0} \frac{15}{01.x0.0x.x0}$$

$$\xrightarrow{v_1=0} \frac{5}{0x.x0.0x.x0}.$$

Scanned with OKEN Scanner

Al 4-lea impuls

$$5 \xrightarrow{x=1} \frac{25}{0x.01.0x.x0} \xrightarrow{v_2=0} \frac{21}{0x.0x.0x.x0} \xrightarrow{x=0} \frac{1}{0x.0x.10.x0}$$

$$\xrightarrow{v_3=1} \frac{3}{0x.0x.x0.x0}.$$

Al 5-lea impuls

$$3 \xrightarrow{x=1} \frac{23}{0x.10.x0.x0} \xrightarrow{v_2=1} \frac{27}{0x.x0.x0.x0} \xrightarrow{x=0} \frac{7}{10.x0.x0.x0}$$

$$\xrightarrow{v_1=1} \frac{17}{x0.x0.x0.x0}.$$

Al 6-lea impuls

$$17 \xrightarrow{x=1} \frac{37}{x0.01.x0.x0} \xrightarrow{v_2=0} \frac{33}{x0.0x.x0.x0} \xrightarrow{x=0} \frac{13}{x0.0x.x0.01}$$

$$\xrightarrow{v_4=0} \frac{12}{x0.0x.x0.0x}.$$

Al 7-lea impuls

$$12 \xrightarrow{x=1} \frac{32}{x0.10.x0.0x} \xrightarrow{v_2=1} \frac{36}{x0.x0.x0.0x} \xrightarrow{x=0} \frac{16}{01.x0.x0.0x}$$

$$\xrightarrow{v_1=0} \frac{6}{0x.x0.x0.0x}.$$

Scanned with OKEN Scanner

Al 8-lea impuls

$$6 \xrightarrow{x=1} \frac{26}{0x.01.x0.0x} \xrightarrow{y_2=0} \frac{22}{0x.0x.x0.0x} \xrightarrow{x=0} \frac{2}{0x.0x.01.0x} \xrightarrow{y_3=0} 0.$$

Stările acestei secvenţe ne permit să structurăm următorul simbol de marcare :

$$F(I_1, S_1, I_2, S_2, I_3, S_3, I_4, S_4) = D^{xy_1 \cdot y_2y_3y_4}$$

I_1 : 1(4, 7), 0 : (0, 1, 2, 3, 5, 6, 15, 16, 20, 21, 22, 23, 24, 25, 26, 27)

S_1 : 1 : (15, 16) 0 : (4, 7, 10, 11, 12, 13, 14, 17, 30, 31, 32, 33, 34, 35, 36, 37)

I_2 : 1 : (20, 23, 31, 32) 0 : (0, 1, 2, 3, 10, 11, 12, 13, 21, 22, 25, 26, 30, 33, 34, 37)

S_2 : 1 : (25, 26, 34, 37) 0 : (4, 5, 6, 7, 14, 15, 16, 17, 20, 23, 24, 27, 31, 32, 36)

I_3 : 1 : (1) 0 : (0, 2, 4, 5, 10, 11, 14, 15, 20, 21, 24, 25, 30, 31, 34, 35)

S_3 : 1 : (2) 0 : (1, 3, 6, 7, 12, 13, 16, 17, 22, 23, 26, 27, 32, 33, 36, 37)

I_4 : 1 : (10) 0 : (0, 2, 4, 6, 12, 13, 14, 16, 20, 22, 24, 26, 30, 32, 34, 36)

S_4 : 1 : (13) 0 : (1, 3, 5, 7, 10, 11, 15, 17, 21, 23, 25, 27, 31, 33, 35, 37)

Expresiile logice minimale care corespund variabilelor definite prin acest simbol, rezultă din :

Pentru I_1

$$4 \,/\, \overset{10}{14} \rightarrow 00.\underline{1}00 \rightarrow \bar{x}y_2\bar{y}_3\bar{y}_4.$$

$$7 \,/\, \overset{10}{17} \rightarrow 0\underline{0}.111 \rightarrow \bar{x}y_2y_3y_4.$$

$$I_1 = \bar{x}y_2(\bar{y}_3\bar{y}_4 \cup y_3y_4)$$

Pentru S_1

$$15 \overset{10}{/} 5 \rightarrow 01.101 \rightarrow \bar{x}y_2\bar{y}_3y_4.$$

$$16 \overset{10}{/} 6 \rightarrow 01.110 \rightarrow \bar{x}y_2y_3\bar{y}_4.$$

$$S_2 = \bar{x}y_2(\bar{y}_3y_4 \cup y_3\bar{y}_4).$$

Pentru I_2

$$20 \overset{4}{/} 24 \overset{20}{/} 0\ 4 \rightarrow 10.000 \rightarrow \bar{y}_1\bar{y}_3\bar{y}_4.$$

$$23 \overset{4}{/} 27 \rightarrow 10.011 \rightarrow x\bar{y}_1y_3y_4.$$

$$31 \overset{4}{/} 35 \rightarrow 11.001 \rightarrow xy_1\bar{y}_3y_4.$$

$$32 \overset{4}{/} 36 \rightarrow 11.010 \rightarrow xy_1y_3\bar{y}_4.$$

$$I_2 = \bar{y}_1\bar{y}_3\bar{y}_4 \cup x\,(\bar{y}_1y_3y_4 \cup y_1\bar{y}_3y_4 \cup y_1y_3\bar{y}_4).$$

Pentru S_2

$$25 \overset{4}{/} 21 \rightarrow 10.101 \rightarrow x\bar{y}_1\bar{y}_3y_4.$$

$$26 \overset{4}{/} 22 \rightarrow 10.110 \rightarrow x\bar{y}_1y_3\bar{y}_4.$$

$$34 \overset{1}{/} 35 \rightarrow 11.101 \rightarrow xy_1y_2\bar{y}_3.$$

Scanned with OKEN Scanner

$$37 \overset{2}{/} 35 \rightarrow 11.1\underline{1}1 \rightarrow xy_1y_2y_4.$$

$$S_2 = x(\overline{y}_1\overline{y}_3y_4 \cup \overline{y}_1y_3\overline{y}_4 \cup y_1y_2\overline{y}_3 \cup y_1y_2y_4).$$

Pentru I_3

$$1 \overset{2}{/} 3 \rightarrow 00.0\underline{0}1 \rightarrow I_3 = \bar{x}\overline{y}_1\overline{y}_2y_4.$$

Pentru S_3

$$2 \overset{2}{/} 0 \rightarrow 00.0\underline{1}0 \rightarrow S_3 = \bar{x}\overline{y}_1\overline{y}_2\overline{y}_4.$$

Pentru I_4

$$10 \overset{1}{/} 11 \rightarrow 01.00\underline{0} \rightarrow I_4 = \bar{x}y_1\overline{y}_3\overline{y}_2.$$

Pentru S_4

$$13 \overset{1}{/} 12 \rightarrow 01.00\underline{1} \rightarrow S_4 = \bar{x}y_1\overline{y}_2y_3.$$

Expresiile logice găsite permit structurarea schemei numărătorului de 7.

Exemplul 11.30. Să se facă sinteza schemei unei celule dintr-un registru de deplasare prevăzută cu două borne de intrare şi o bornă de ieşire.

La intrarea x se aplică semnalul ce vine de la celula precedentă, semnal ce urmează a fi elaborat la ieşirea Z în tactul următor.

La intrarea t, se aplică impulsurile de tact care determină deplasarea în registru.

Semnalul de ieşire reproduce după un tact semnalul aplicat la intrarea x.

Problema a fost inspirată din lucrarea [91 p. 405].

Scanned with OKEN Scanner

Circuitul elementar al registrului de deplasare este caracterizat prin următoarele stări : starea iniţială, starea care succede aplicării primului semnal de tact în prezenţa semnalului x, starea care succede aplicării celui de al doilea semnal în prezenţa sau în absenţa semnalului x şi starea ce succede dispariţiei celui de al doilea semnal de tact.

Pentru memorarea acestor patru stări sînt necesare două variabile secundare, astfel încît putem structura următorul raport de stare :

$$\frac{t \cdot xy_1y_2}{ZY_1Y_2}.$$

Presupunem că semnalul x se aplică fie la începutul semnalului de tact, fie pe durata acestui semnal. În acest caz, schema trece prin stările

$$\frac{0}{0} \xrightarrow{t=1} \frac{10}{0} \xrightarrow{x=1} \frac{14}{0} \xrightarrow{t=0} \frac{4}{2} \xrightarrow{v_1=1} \frac{6}{2}.$$

Dacă semnalul x nu dispare decît după apariţia celui de-al doilea semnal de tact, atunci schema trece prin stările :

$$6 \xrightarrow{t=1} \frac{16}{3} \xrightarrow{v_2=1} \frac{17}{7} \xrightarrow{x=0} \frac{13}{7} \xrightarrow{t=0} \frac{3}{5} \xrightarrow{v_1=0} \frac{1}{5}.$$

O dată cu apariţia celui de-al treilea tact, schema revine în starea iniţială, astfel încît putem scrie :

$$1 \xrightarrow{t=1} \frac{11}{4} \xrightarrow{v_2=0} 10.$$

Stările definite prin secvenţa de mai sus permit structurarea următorului simbol de marcare :

$$F(Z, Y_1, Y_2) = D_Z^{t \cdot xy_1y_2} : 1 : (1, 3, 11, 13, 17)\ 0 : (0, 4, 6, 10, 14, 16)$$
$$Y_1 : 1 : (4, 6, 13, 16, 17)\ 0 : (0, 1, 3, 10, 11, 14)$$
$$Y_2 : 1(1, 3, 12, 16, 17)\ 0 : (0, 4, 6, 10, 11, 14).$$

Scanned with OKEN Scanner

Urmărind structurarea unei scheme modulare UNILOG, vom folosi metoda de minimizare prin grupe de 0. Astfel, din simbolul de marcare dedus rezultă :

Pentru Z

$$\begin{array}{l} \overset{2}{0} \,/\, \overset{4}{2} \,/\, 4\ \overset{10}{6} \,/\, 10\ 12\ 14\ 16 \end{array} \rightarrow \underline{0}.\underline{000} \rightarrow \overline{y}_2.$$

Pentru Y_1

$$\begin{array}{l} \overset{1}{0} \,/\, \overset{2}{1} \,/\, 2\ 3 \end{array} \rightarrow 0.\underline{000} \rightarrow \overline{t}\overline{x}.$$

Pentru Y_1 *și* Y_2

$$\begin{array}{l} \overset{1}{10} \,/\, \overset{4}{11} \,/\, 14\ 15 \end{array} \rightarrow 1.\underline{00}\underline{0} \rightarrow t\overline{y}_1.$$

Pentru Y_2

$$\begin{array}{l} \overset{2}{4} \,/\, \overset{4}{6} \,/\, 0\ 2 \end{array} \rightarrow 0.\underline{1}00 \rightarrow \overline{t}\overline{y}_2.$$

Pentru eliminarea hazardului de comutație se mai impune formarea unor grupe care să conțină cel puțin cîte un număr de stare comun grupărilor care caracterizează fiecare dintre variabilele de ieșire.

Se aleg grupările :

Pentru Y_1

$$\begin{array}{l} \overset{10}{0} \,/\, \overset{1}{10} \,/\, 1\ 11 \end{array} \rightarrow \underline{0}.\underline{000} \rightarrow (\overline{x}\overline{y}_1)$$

Pentru Y_2

$$\begin{array}{l} \overset{10}{0} \,/\, \overset{4}{10} \,/\, 4\ 14 \end{array} \rightarrow \underline{0}.\underline{000} \rightarrow (\overline{y}_1\overline{y}_2).$$

Se obțin *astfel expresiile* logice minimale :

$$Z = y_2;\quad Y_1 = \overline{t\bar{y}_1 \cup \bar{t}\bar{x} \cup (\bar{x}\bar{y}_1)} = (\bar{t} \top y_1) \top (t \top x) \top (x \top y_1).$$

$$Y_2 = \overline{t\bar{y}_1 \cup \bar{t}\bar{y}_2 \cup (\bar{y}_1\bar{y}_2)} = (\bar{t} \top y_1) \top (t \top y_2) \top (y_1 \top y_2).$$

Schema modulară UNILOG care corespunde acestor formule este dată în figura 11.66.

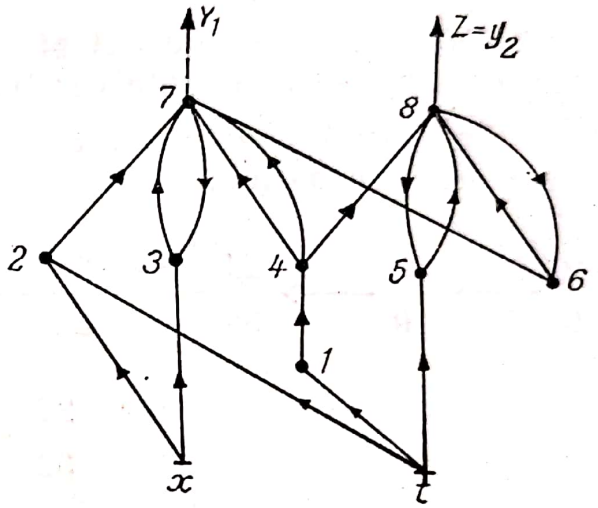


Fig. 11.66. — Element dintr-un registru de deplasare.

Exemplul 11.31. Ne propunem să facem sinteza unui generator de tacte, adică a unui dispozitiv cu ajutorul căruia se obține un tren de impulsuri dreptunghiulare echidistante, la care durata semnalului și a pauzei este constantă în timp (fig. 11.67).

Se presupune că pentru măsurarea duratelor τ_1 și τ_2 se dispune de un modul „TIMP", care funcționează după diagrama dată în figura 11.68.

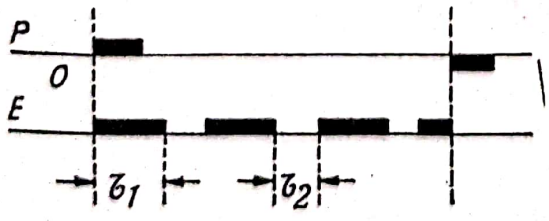


Fig. 11.67. — Ciclograma generatorului de tacte.

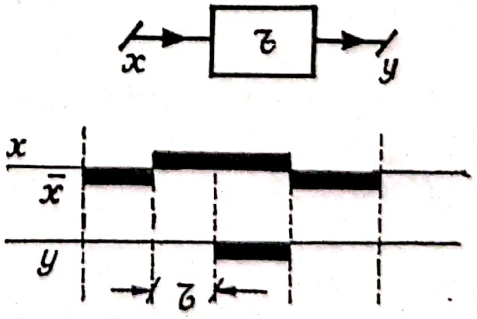


Fig. 11.68. — Diagrama elementului TIMP.

Considerînd P, butonul de comandă a pornirii generatorului de tacte, T_1 şi T_2 cele două variabile de ieşire ataşate elementelor „TIMP" folosite şi E, variabila de ieşire, se structurează raportul de stare

$$\frac{ab \cdot t_1 t_2 P}{EA \cdot BT_1 T_2}.$$

Variabilele auxiliare A şi B au fost introduse pentru memorarea stărilor schemei pe durata existenţei şi lipsei semnalului de ieşire.

În formă codificată, secvenţa de lucru a generatorului de tacte se scrie astfel:

$$\frac{0}{0} \xrightarrow{P=1} \frac{1}{10} \xrightarrow{a=1} \frac{21}{32} \xrightarrow{P=0} \frac{20}{32} \xrightarrow{t_1=1} \frac{24}{37} \xrightarrow{b=1} \frac{34}{15} \xrightarrow{t_1=0} \frac{30}{15} \xrightarrow{t_2=1} \frac{32}{11}$$

$$\xrightarrow{b=0} \frac{22}{32} \xrightarrow{t_2=0} 20 \ldots$$

Analiza acestei secvenţe ne conduce la structurarea simbolului de marcare:

$$F(E, A, B, T_1, T_2) = D^{ab \cdot t_1 t_2 P}$$
$$E: 1: (20, 21, 22, 24)\ 0: (0, 1, 30\ 32, 34)$$
$$A: 1: (1, 20, 21, 22, 24, 30, 32, 34)\ 0: (0)$$
$$B: 1: (24, 30, 34)\ 0: (0, 1, 20, 21, 22, 32)$$
$$T_1: 1: (20, 21, 22, 24)\ 0: (0, 1, 30, 32, 34)$$
$$T_2: 1\ (24, 30, 32, 34)\ 0: (0, 1, 20\ 21, 22)$$

Rezultă expresiile logice minimale:

Pentru E

$$\begin{array}{c} 1 \quad 2 \qquad 4 \\ 20 \,/\, 21 \,/\, 22\ 23 \,/\, 24\ 25\ 26\ 27 \rightarrow 10.\underline{000} \end{array}$$

$$E = a\bar{b}.$$

Pentru A

$$20\,\frac{1}{21}\,/\,3\,/\,5\;7\,/\,11\;13\;15\;17 \;\;\overset{2\quad 4\quad\quad 10}{\cdot\;\cdot\;\cdot\quad\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot} \rightarrow \underline{00}.\underline{001} \rightarrow P$$

$$20\,\frac{20}{30}\,/\,21\,/\,22\;23\,/\,24\;25\;26\;27 \;\;\overset{1\quad 2\quad\quad 4}{\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot} \rightarrow 1\underline{0}.\underline{000} \rightarrow a.$$

$$A = (a \cup P)\,\overline{O}.$$

Variabila $\overline{O}$, a fost introdusă pentru a ține seama de funcția butonului de oprire O, a generatorului de tacte.

Pentru B

$$20\,\frac{24}{4}\,/\,25\,/\,26\;27\,/\;\;34\;\;35\;\;36\;\;37 \;\;\overset{1\quad 2\quad\quad 10}{\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot} \rightarrow \underline{10}.\underline{100} \rightarrow t_1.$$

$$30\,/\,31\,/\,34\;35\,/\,10\;11\;14\;15 \;\;\overset{1\quad 4\quad\quad 20}{\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot} \rightarrow \underline{11}.\underline{00}\underline{0} \rightarrow b\bar{t}_2.$$

$$B = t_1 \cup b\bar{t}_2 = t_1(\bar{t}_2) \cup b\bar{t}_2 = \bar{t}_2(b \cup t_1).$$

Variabila redondantă $\bar{t}_2$ s-a introdus avînd în vedere structurarea mai ușoară a schemei modulare UNILOG după ultima formă posibilă a expresiei logice.

Pentru T_1

$$20\,/\,21\,/\,22\;23\,/\,24\;25\;26\;27 \;\;\overset{1\quad 2\quad\quad 4}{\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot\;\cdot} \rightarrow 10.\underline{000}$$

$$T_1 = a\bar{b}.$$

Scanned with OKEN Scanner

Pentru T_2

$$20\begin{matrix}24\\4\end{matrix} \begin{matrix}1\\/\\.\end{matrix} 25 \begin{matrix}2\\/\\.\end{matrix} 26\ 27 \begin{matrix}10\\/\\.\end{matrix} 34\ 35\ 36\ 37 \rightarrow \underline{10.100} \rightarrow t_1.$$

$$20\begin{matrix}30\\10\end{matrix} \begin{matrix}1\\/\\.\end{matrix} 31 \begin{matrix}2\\/\\.\end{matrix} 32\ 33 \begin{matrix}4\\/\\.\end{matrix} 34\ 35\ 36\ 37 \rightarrow \underline{11.000} \rightarrow b.$$

$$T_2 = b \cup t_1 = \overline{\overline{b}\,\overline{t}_1}.$$

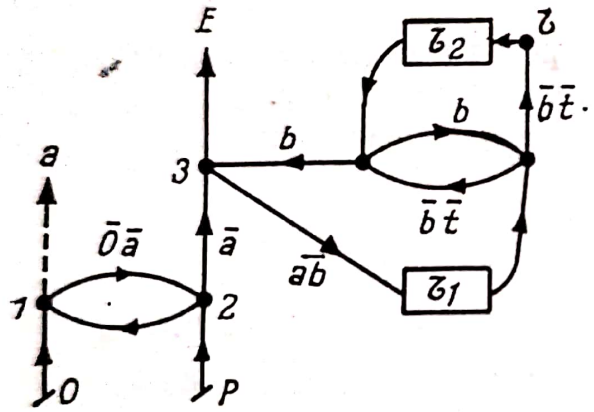


Fig. 11.69. — Schema generatorului de tacte.

Schema modulară UNILOG a generatorului de tacte este dată în figura 11.69. Ea a fost realizată practic la Laboratorul de electronică a Institutului politehnic Iaşi şi a dat rezultate bune.

11.4. PROIECTAREA LOGICĂ A SCHEMEI DE AUTOMATIZARE A UNUI LAMINOR DE BUMBAC

11.4.1. SPECIFICAŢIILE SCHEMEI DE AUTOMATIZARE

Presupunem că laminorul este acţionat de un electromotor M, protejat la suprasarcini mari de scurtă durată prin siguranţele fuzibile S, şi la suprasarcini mici, dar de lungă durată, prin releele termice R_{tk}. Contactele lor vor fi notate prin r_{tk}.

Conectarea manuală a motorului la rețeaua electrică trifazată se face prin intermediul unui heblu tripolar H iar pornirea automată a lui este asigurată prin contactele c_m, ale unui contactor C_m.

Schema monofilară a circuitului de forță a electromotorului este dată în figura 11.70.

Cuplarea motorului de acționare M cu laminorul se face prin intermediul unei cuple electromagnetice E.

Pentru acționarea căruciorului port-căni se folosește un servomotor pneumatic S_t (fig. 11.71) alimentat prin amplificatorul pneumatic bistabil A_t, care este comandat prin ventilele electromagnetice V_s și V_d, corespunzătoare sensului de deplasare respectiv spre stînga sau dreapta a ghiarei ce antrenează căruciorul port-căni.

Pentru ridicarea discului port-cană se folosește un alt servomotor pneumatic S_d, monostabil, alimentat prin amplificatorul pneumatic A_d, care este comandat prin ventilul electromagnetic V_c. Dacă bobina acestui ventil este străbătută de curent ($V_c = 1$), ventilul V_c este deschis și drept consecință discul port-cană se găsește în poziția inferioară ($D = 1$). Dacă $V_c = 0$, discul se găsește în poziția superioară ($D = 0$).

Controlul cursei discului port-cană se face prin micro-întrerupătorul (limitator de cursă) D, iar controlul deplasării ghiarei de antrenare a căruciorului port-căni se face prin microîntrerupătoarele T_s și T_d.

Funcțiile îndeplinite de schema de automatizare a laminorului sînt:

a) Memorarea comenzilor de pornire sau oprire a servomotorului M, comenzi date prin apăsarea butoanelor cu revenire P_m și O_m.

b) Memorarea comenzilor de pornire sau oprire a laminorului (conectarea sau deconectarea cuplei electromagnetice E), comenzi posibil de dat din mai multe locuri prin apăsarea butoanelor cu revenire P_{Mk} și O_{Mk}.

c) Pornirea sau oprirea laminorului (cupla E), fără memorarea comenzii, prin comenzi posibil de dat din mai multe locuri prin apăsarea butoanelor cu revenire P_{ik} (pornire intermitentă pentru probe tehnologice). Această pornire trebuie să poată fi făcută indiferent dacă există sau nu există defect ($C_{ik} = 1$).

d) Schimbarea automată a căniі la atingerea lungimii programate a benzii laminate sau la o comandă dată manual. Controlul

Scanned with OKEN Scanner

lungimii benzii laminate se face printr-un contact al unui contor de lungime C_l iar comanda manuală se dă prin apăsarea unui buton cu revenire T_m (transport manual). În cazul cînd T_m se menține apăsat, este pregătită schimbarea cănii, care poate avea loc numai după ce T_m nu mai este apăsat ($T_m = 0$).

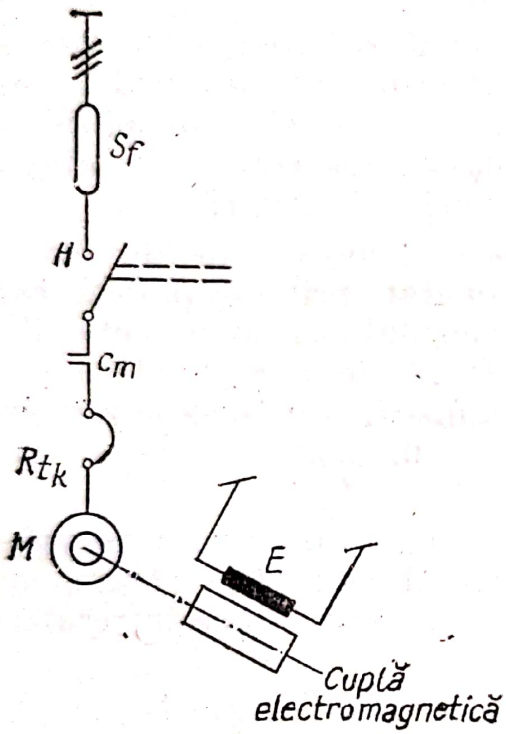
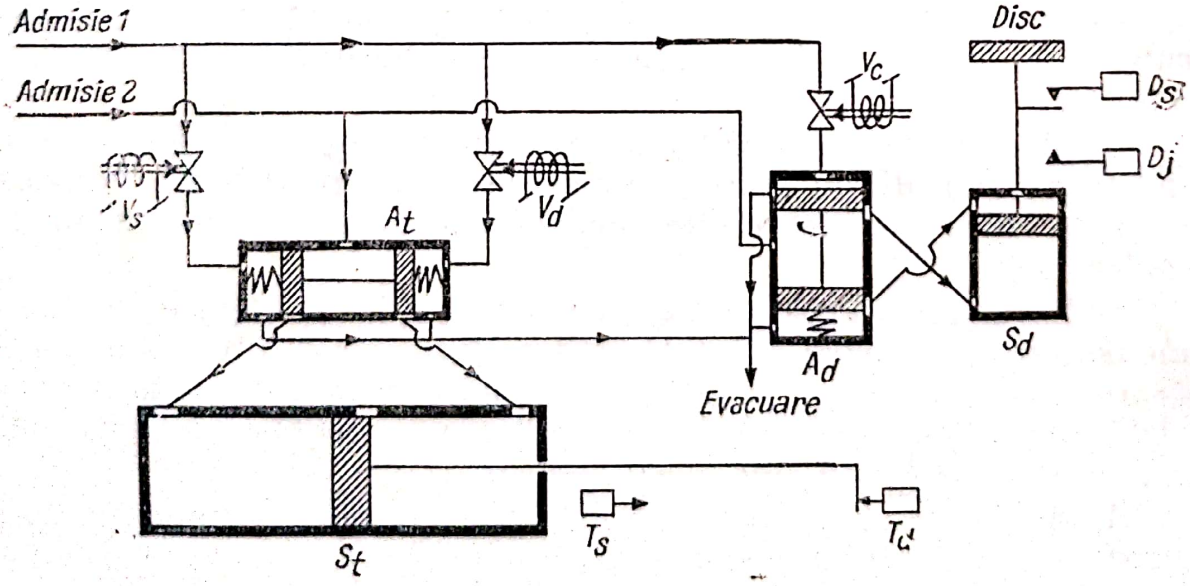


Fig. 11.70. — Schema de alimentare a motorului de acționare a laminorului de bumbac.

Fig. 11.71. — Schema de acționare pneumatică a schimbătorului automat de cană.



e) Posibilitatea de a comanda manual coborîrea discului port-cană prin comutarea unui comutator bipoziţional P_d şi posibilitatea interzicerii deplasării căruciorului port-căni prin comutarea unui alt comutator bipoziţional P_t.

f) Controlul automat al continuităţii benzii de bumbac la rastelul de alimentare din stînga, respectiv dreapta, al continuităţii benzii laminate la cilindrii de laminare şi a intrării benzii laminate în cană. Acest control este făcut prin contactele tehnologice, notate respectiv prin C_s, C_d, C_e şi C_b. Intervenţia oricăruia dintre aceste contacte este memorată astfel încît revenirea în stare normală a acestor contacte, sau comanda de pornire a laminorului dată de o altă persoană decît cea care efectuează această remediere, să nu aibă nici un efect. Laminorul poate fi repus în funcţiune numai după ce se şterge memoria prin apăsarea butonului de oprire O_{Mk} şi apoi se dă comanda de pornire prin apăsarea pe P_{Mk}.

g) Posibilitatea de funcţionare neautomată a laminorului, cînd supravegherea funcţionării lui este făcută de un operator uman. În acest scop se foloseşte o cheie bipoziţională M_a, una dintre poziţii corespunzînd funcţionării automate şi alta, funcţionării neautomate.

h) Semnalizarea optică a prezenţei canei de rezervă, a stării de funcţionare a schemei de comandă, a stării de repaos a laminorului şi a cauzei ce a determinat oprirea lui automată. În acest scop se vor folosi respectiv lămpile semnalizatoare de tablou : L_p, L_m, L_s, L_d, L_e şi L_b şi o lampă cu lumină roşie L_r, aşezată în partea superioară a maşinii astfel încît să poată fi observată de la distanţă de către personalul de deservire.

i) Asigurarea securităţii depanatorului, pe durată cît efectuează reparaţii sau reglaje la cilindrii de laminare. În acest scop se foloseşte un contact al blocării de securitate B_s, a cărui deschidere are loc odată cu luarea capacului protector de acces la cilindrii de laminare.

j) Asigurarea blocării tuturor comenzilor, pe durata de repaos comandat a laminorului. În acest scop se foloseşte o cheie de siguranţă K_s, posibil de acţionat numai cu ajutorul cheii de acces la maşină, cheie purtată de persoana ce are în sarcină exploatarea laminorului.

k) Blocarea comenzii motorului M prin releele termice şi prin releul de presiune R_p, folosit la instalaţia pneumatică ce deserveşte laminorul astfel încît acesta să nu poată fi acţionat decît dacă presiunea aerului din instalaţia pneumatică are valoarea programată.

Scanned with OKEN Scanner

11.4.2. CODIFICAREA SPECIFICAȚIILOR SCHEMEI DE AUTOMATIZARE

Funcționarea automată a laminorului poate fi descrisă printr-o succesiune de stări, succesiune care reprezintă secvența de lucru a lui. Fiecare stare a laminorului se exprimă în formă codificată printr-un raport, numit rapot de stare, așa cum s-a arătat mai sus.

Analizînd specificațiile schemei de automatizare, rezultă următorul grup al variabilelor de intrare :

$$r_{tk}c_m D T_s T_d P_m O_m P_{Mk} O_{Mk} P_{ik} C_l T_m P_d P_t C_p C_{tk} M_a B_s K_s R_p$$

unde prin C_{tk} s-a notat o variabilă generică atașată unui contact de blocare tehnologică iar prin C_p s-a notat traductorul cu contact sesizor al prezenței canei de rezervă.

În scopul simplificării procesului de proiectare, vom omite din acest șir variabilele atașate contactelor ce determină oprirea de urgență a motorului M și anume variabilele :

$$K_s O_m R_p r_{tk},$$

variabilele ce determină oprirea de urgență a laminorului (decuplarea cuplei E), și anume variabilele : K_s, O_{Mk} și B_s și variabilele C_p și C_l. De funcțiile logice ale acestor variabile se va ține cont în formulele finale după care se structurează schema de automatizare. După cum se va vedea din cele ce urmează, este necesară și introducerea a două variabile secundare r_k și r_1.

Cu aceste precizări, vom lua în considerare următorul grup al variabilelor de intrare :

$$r_k r_1 \cdot c_m D T_s \cdot T_d P_m P_{Mk} \cdot P_{ik} T_m P_d \cdot P_t C_{tk} M_a .$$

Variabilele de ieșire, exprimă în formă binară starea bobinelor de excitație ale releelor, contactoarelor, electromagneților, lămpilor semnalizatoare, etc., adică a aparatelor comandate prin circuitele schemei de automatizare.

Din analiza specificațiilor schemei de automatizare, rezultă următorul grup al variabilelor de ieșire :

$$R_k \cdot R_1 C_m E \cdot V_s V_d V_c$$

unde, de asemenea, din motive ce se vor vedea ulterior, s-au adăugat variabilele de ieşire auxiliare R_k şi R_1.

Lămpile semnalizatoare sînt legate funcţional de aceste variabile de ieşire, astfel încît :

$$L_m = C_m; \qquad L_{tk} = R_k; \qquad L_r = \overline{E}; \qquad L_p = C_p.$$

Rezultă deci următorul raport de stare generic al schemei de automatizare a laminorului :

$$\left(\frac{r_k r_1 \cdot c_m D T_s \cdot T_d P_m P_{Mk} \cdot P_{ik} T_m P_d \cdot P_t C_{tk} M_a}{R_k \cdot R_1 C_m E \cdot V_s V_d V_c}\right)_n.$$

unde n, este un număr de ordine dat stării respective. Astfel, starea $n = 0$, va fi descrisă codificat prin raportul de stare

$$\left(\frac{1.000}{0}\right)_0.$$

Stările secvenţei de lucru a laminorului, care corespund unor combinaţii diferite ale valorilor binare, momentane, ale variabilelor de ieşire, trebuie să fie distincte, adică trebuie să fie descrise prin numere de stare diferite. Adesea, pentru discriminarea lor, este necesară introducerea variabilelor secundare, ataşate fie variabilelor de ieşire care descriu starea unor relee sau contactoare, fie unor noi variabile de ieşire, care sînt necesar a fi introduse şi care corespund unor relee auxiliare. Introducerea acestor variabile se face în timpul stabilirii secvenţei de lucru a schemei de automatizare, în forma sa codificată, în măsura în care acest lucru este impus de necesitatea de a discrimina stări diferite ale acestei secvenţe.

Variabilele secundare se adaugă capătului din stînga al şirului variabilelor de intrare şi se notează cu literele mici, corespunzătoare variabilelor de ieşire cărora le sînt ataşate. În acest fel, numerele octale, cuprinse de stările anterioare ale secvenţei, nu sînt afectate prin introducerea noilor variabile.

Scanned with OKEN Scanner

Avînd stabilit raportul de stare, putem trece la stabilirea în formă codificată a secvenţei de lucru a laminorului. Schematic, aceasta se prezintă astfel :

Specificaţia a)

$$\left(\frac{1.000}{0}\right)_0 \xrightarrow{P_m=1} \left(\frac{1.200}{20}\right)_1 \xrightarrow{c_m=1} \left(\frac{5.200}{20}\right)_2 \xrightarrow{P_m=0} \left(\frac{5.000}{20}\right)_3$$

Specificaţia b)

$$5.000 \xrightarrow{P_{Mk}=1} \left(\frac{5.100}{60}\right)_4 \xrightarrow{r_1=1} \left(\frac{15.100}{70}\right)_5 \xrightarrow{P_{Mk}=0} \left(\frac{15.000}{70}\right)_6$$

Pentru memorarea comenzii date prin P_{Mk}, a fost necesară introducerea variabilei secundare r_1 şi a celei de ieşire, auxiliare R_1.

Specificaţia c)

În regim normal, avem :

$$5.000 \xrightarrow{P_{ik}=1} \left(\frac{5.040}{30}\right)_7 \xrightarrow{P_{ik}=0} 5.000$$

În regim de avarie, avem :

$$5.000 \xrightarrow{C_{tk}=1} \left(\frac{5.002}{120}\right)_8 \xrightarrow{r_k=1} \left(\frac{25.002}{120}\right)_9 \xrightarrow{P_{ik}=1} \left(\frac{25.042}{130}\right)_{10} \xrightarrow{P_{ik}=0} 25.002$$

Memorarea comenzii date prin C_{tk} impune adăugarea variabilei secundare r_k şi a celei de ieşire, căreia îi este asociată, R_k.

Specificaţia d)

$$15.000 \xrightarrow{T_m=1} \left(\frac{15.020}{71}\right)_{11} \xrightarrow{D=1} \left(\frac{17.020}{33}\right)_{12} \xrightarrow{r_1=0} \left(\frac{7.020}{33}\right)_{13}$$

$$\xrightarrow{T_m=0} \left(\frac{7.000}{33}\right)_{14} \xrightarrow{T_s=0} \left(\frac{6.000}{33}\right)_{15} \xrightarrow{T_d=1} \left(\frac{6.400}{73}\right)_{16} \xrightarrow{r_1=1} \left(\frac{16.400}{75}\right)_{17}$$

$$\xrightarrow{T_d=0} \left(\frac{16.000}{75}\right)_{18} \xrightarrow{T_s=0} \left(\frac{17.000}{70}\right)_{19} \xrightarrow{D=0} 15.000$$

Scanned with OKEN Scanner

Dacă T_m rămîne apăsat, avem starea de blocare:

$$16.400 \xrightarrow{T_m=1} \left(\frac{16.420}{71}\right)_{20}$$

Specificația e)

$$5.000 \xrightarrow{P_d=1} \left(\frac{5.010}{21}\right)_{21} \xrightarrow{D=1} \left(\frac{7.010}{21}\right)_{22}$$

$$15.000 \xrightarrow{P_d=1} \left(\frac{15.010}{71}\right)_{23} \xrightarrow{D=1} \left(\frac{17.010}{71}\right)_{24}$$

$$16.400 \xrightarrow{P_t=1} \left(\frac{16.404}{71}\right)_{25} \text{ și : } 16.420 \xrightarrow{P_t=1} \left(\frac{16.424}{71}\right)_{26}$$

Specificația f)

$$25.002 \xrightarrow{C_{tk}=0} \left(\frac{25.000}{120}\right)_{27} \xrightarrow{P_{Mk}=1} \left(\frac{25.100}{120}\right)_{28}$$

$$15.000 \xrightarrow{C_{tk}=1} \left(\frac{15.002}{170}\right)_{29} \xrightarrow{r_k=1} \left(\frac{35.002}{130}\right)_{30} \xrightarrow{r_1=0} \left(\frac{25.002}{120}\right)_{31}$$

Specificația g)

$$5.000 \xrightarrow{M_a=1} \left(\frac{5.001}{20}\right)_{32} \xrightarrow{C_{tk}=1} \left(\frac{5.002}{20}\right)_{33}$$

$$15.000 \xrightarrow{M_a=1} \left(\frac{15.001}{70}\right)_{34} \xrightarrow{C_{tk}=1} \left(\frac{15.003}{70}\right)_{35}.$$

De restul specificațiilor schemei de automatizare se ține cont numai la structurarea formulelor finale care vor servi la structurarea acestei scheme.

11.4.3. STRUCTURAREA SIMBOLURILOR DE MARCARE

Cînd numărul variabilelor de intrare este mare, avînd în vedere că fiecare variabilă de ieșire, luată separat, nu depinde adesea de totalitatea variabilelor de intrare, se poate structura cîte un simbol de marcare separat, pentru fiecare variabilă de ieșire.

Printre variabilele de intrare de care depinde o variabilă de ieşire, luată separat, se vor găsi în primul rînd variabilele care modifică valoarea binară a acestei variabile. Ele pot fi uşor depistate inspectînd stările secvenţelor de lucru ale schemei.

Din restul variabilelor de intrare, se adaugă numărul minim necesar de variabile, necesare pentru discriminarea secvenţei particulare de lucru, aferente variabilei respective de ieşire. Se obţine astfel şirul variabilelor de intrare de care depinde variabila de ieşire considerată, şir care reprezintă indicele superior al simbolului de marcare aferent acesteia.

Vom exemplifica procedeul folosit pentru variabila de ieşire R_1, reprezentată codificat în numitorul raportului de stare prin cifra de pondere 40.

Inspectînd cele 35 stări ale secvenţelor de lucru ale schemei, constatăm că variabila R_1 îşi modifică valoarea în cazul tranziţiilor între stările : (3—4), (11—12), (15—16) şi (29—30), tranziţii determinate de intervenţia variabilelor : P_{Mk}, d, T_d şi r_k, ale căror ponderi formează numărul octal 22.500.

Folosind numai cifrele acestui număr, celei 8 stări de mai sus se scriu :

$$R_1 = 0 : (0)_3,\ (2.000)_{12},\ (2.000)_{15},\ (20.000)_{30}$$

$$R_1 = 1 : (100)_4,\ (0)_{11},\ (2.400)_{16},\ (0)_{29}.$$

Observăm că se impune discriminarea stării 3 de stările 11 şi 29. Adăugarea varabilei r_1, de pondere 10.000 şi a variabilei T_m, de pondere 20, rezolvă problema. În acest caz, folosind numai cifrele numărului 32.520,cele 35 stări ale secvenţelor de lucru ale schemei se reduc la :

$$R_1 = 0 : 0,20.000,\ 12.020,\ 2.020,\ 2.000,\ 20.100,\ 30.000.$$

$$R_1 = 1 : 100,\ 10.100,\ 10.000,\ 10.020,\ 2.400,\ 12.400,\ 12.000,\ 2.420.$$

Numerele de stare cuprinse de indicele inferior al simbolului de marcare se obţin efectuînd în stările de mai sus următoarele transformări de ponderi :

$$20.000 \rightarrow 40,\ 10.000 \rightarrow 20,\ 2.000 \rightarrow 10,\ 400 \rightarrow 4,\ 100 \rightarrow 2$$

$$\text{şi : } 20 \rightarrow 1.$$

Scanned with OKEN Scanner

Noile numere de stare se referă la șirul : $r_k r_1 D \cdot T_d P_{Mk} T_m$.
Se obține astfel :

$$R_1 = \overline{K}_s \cdot D_{1:(2,14,15,20,21,22,34)\ 0:(0,10,11,31,40,42,60)}^{r_k r_1 D \cdot T_d P_{Mk} T_m}.$$

Variabila $\overline{K}_s$ a fost introdusă în scopul de a ține seama de rolul de întrerupător general al acestei chei, potrivit specificațiilor date prin tema de proiectare.

Folosind același procedeu, obținem :

$$R_k = \overline{K}_s \overline{O}_{Mk} \cdot D_{1:(2,4,6)\ 0:(0,1,3)}^{r_k C_{tk} M_a}$$

$$C_m = \overline{K}_s \overline{O}_m \overline{R}_p \overline{r}_{tk} \cdot D_{1:(123)\ 0:(0)}^{c_m P_m}$$

$$E = \overline{K}_s \overline{O}_{Mk} \overline{B}_s \cdot D_{1:(2,4,10,11,14,15,22,30)\ 0:(0,1,5,20)}^{r_k r_1 \cdot D P_{ik} P_d}$$

$$V_s = \overline{K}_s \overline{O}_{Mk} \cdot D_{1:(10)\ 0:(0,4,6,11,12,13,14,16)}^{r_1 \cdot T_s T_m P_t}$$

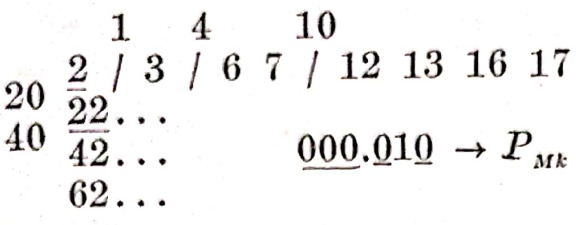
$$V_d = \overline{K}_s \overline{O}_{Mk} \cdot D_{1:(0,14,16,36)\ 0:(4,5,15,20,22,24,25,34,35)}^{r_1 D \cdot T_s T_m P_d}$$

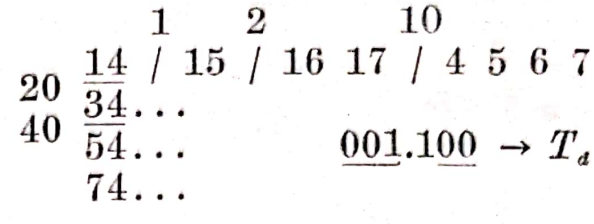
$$V_c = \overline{K}_s \overline{O}_{Mk} \cdot D_{\substack{1:(5,10,14,15,16,25,26,30,32,35,36) \\ 0:(4,24,34).}}^{r_1 D \cdot T_s T_m P_d}$$

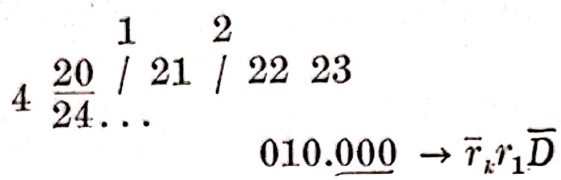
11.4.4. DEDUCEREA EXPRESIILOR LOGICE MINIMALE

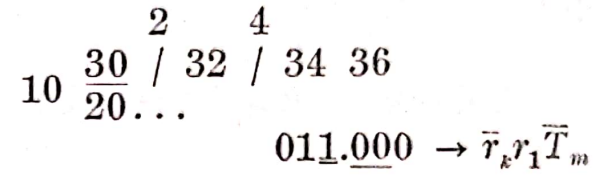
Vom da schema de calcul folosită numai pentru calculul implicanților aferenți simbolului de marcare al variabilei R_1.

Schematic, acest calcul se prezintă astfel :









Circuitul de comutaţie a lui R_1 este lipsit de hazard de comutaţie întrucît legătura între grupe este asigurată prin cel puţin un număr de stare comun.

Rezultă deci

$$R_1 = \overline{K}_s[P_{Mk} \cup T_d \cup \bar{r}_k r_1(\overline{D} \cup \overline{T}_m)].$$

Similar, se deduc următoarele expresii logice ale schemei de automatizare :

$$R_k = \overline{K}_s \overline{O}_{Mk} \overline{M}_a (r_k \cup C_{tk}).$$

$$C_m = \overline{K}_s \overline{O}_m \overline{R}_p \bar{r}_{tk} (P_m \cup c_m).$$

$$E = \overline{K}_s \overline{O}_{Mk} \overline{B}_s (r_1 \cup P_{ik} \cup D\overline{P}_d).$$

$$V_s = \overline{K}_s \overline{O}_{Mk} r_1 \overline{T}_s \overline{T}_m \overline{P}_t.$$

$$V_d = \overline{K}_s \overline{O}_{Mk} [\bar{r}_1 \overline{T}_s \cup D(\bar{r}_1 \overline{P}_d \cup T_m)].$$

$$V_c = \overline{K}_s \overline{O}_{Mk} (P_a \cup \bar{r}_1 D \cup T_m \cup \overline{T}_s).$$

Din specificaţiile schemei de automatizare mai rezultă formulele :

$$T_m = (T_m \cup C_1), \quad L_m = C_m, \quad L_r = \overline{E} = \bar{r}_E, \quad L_{tk} = R_k \text{ şi } L_p = C_p.$$

11.4.5. STRUCTURAREA SCHEMEI DE COMUTAȚIE DINAMICĂ

Pe baza expresiilor logice de mai sus se poate structura schema de comutație dinamică sau statică care satisface specificațiile cerute prin tema de proiectare.

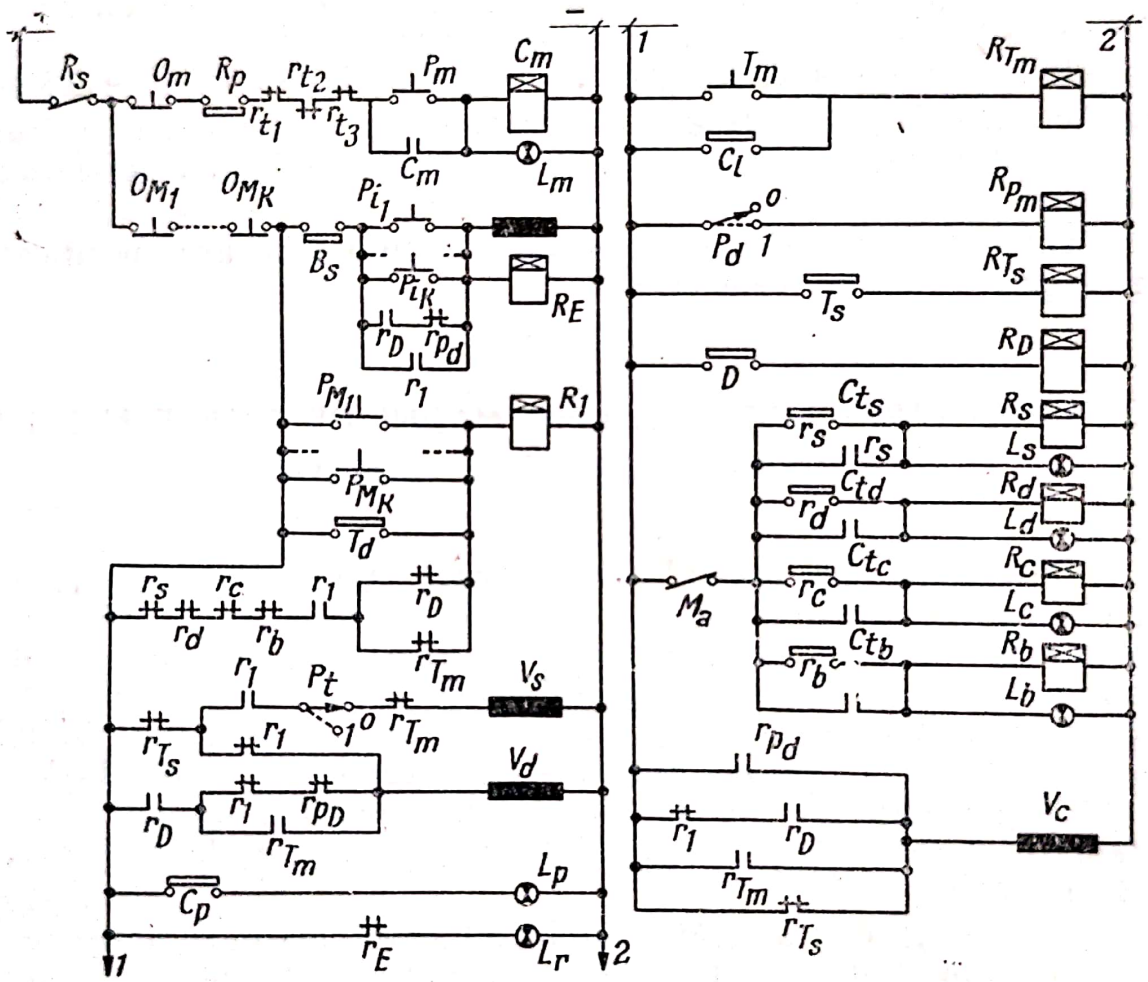


Fig. 11.72. — Schema de comutație dinamică pentru comanda automată a unui laminor de bumbac.

În figura 11.72, este dată schema de comutație dinamică.

Avînd în vedere că microîntrerupătoarele sau butoanele de comandă se execută fie sub forma de contacte unice, fie sub forma asociației dintre un contact normal deschis și unul normal închis, pentru structurarea schemei se impune folosirea unor relee inter-

mediare, comandate de microîntrerupătoarele sau butoanele ce au mai multe contacte folosite în această schemă. Acesta e cazul microîntrerupătoarelor T_s și D, al butonului T_m, legat în derivație cu contactul C_l al contorului de lungime și al comutatorului P_d, pentru care s-au folosit releele intermediare R_{T_s}, R_D, R_{T_m} și R_{P_d}.

În schemă, contactele r_k, C_{tk} și releul R_k, ia valori particulare pentru cele patru blocări folosite. Cele patru circuite de blocare folosesc în comun contactul cheii M_a.

Pentru semnalizarea stării de repaos a laminorului se folosește un releu auxiliar R_E, branșat în derivație pe electromagnetul cuplei E. Contactul acestuia r_E, controlează circuitul lămpii de semnalizare L_r.

În final se verifică dacă schema dedusă satisface specificațiile cerute prin tema de proiectare.

11.5. PROIECTAREA LOGICĂ A UNEI MAȘINI DE GĂURIT AUTOMATE (AUTOMAT SECVENȚIAL ASINCRON)

Literatura de specialitate apărută pînă în prezent oferă proiectantului un număr de metode de proiectare logică a automatelor secvențiale asincrone care devin extrem de laborioase pe măsură ce crește numărul variabilelor care descriu comportarea automatului. Aceste automate au o foarte largă răspîndire în industrie. Încercăm în cele ce urmează să prezentăm o metodă bazată pe noțiunile cuprinse în această lucrare, metodă pe care o considerăm mai expeditivă în raport cu restul metodelor cunoscute pînă în prezent.

În scopul de a arăta aplicabilitatea metodei propuse am reluat exemplul dat de P. Girard în cele două lucrări publicate [40], [41]. Iată enunțul temei de proiectare care face obiectul acestor lucrări.

Se cere structurarea schemei de comandă automată a unei mașini de găurit (fig. 11.73), care să satisfacă următoarele specificații :

a) Piesele de perforat sînt debitate prin culoarul B. În momentul cînd unul dintre capetele tijei presoare A, atinge limitatorul A_1, celălalt capăt eliberează extremitatea culoarului B, astfel încît una dintre piesele de găurit cade în fața tijei presoare (împingătoare).

Prezența ei este semnalată prin semnalul B_1 care inițiază comanda deplasării tijei împingătoare A spre dreapta. Această comandă nu se modifică în cursul deplasării tijei împingătoare care determină dispariția semnalului B_1 ca urmare a împingerii piesei către platanul C.

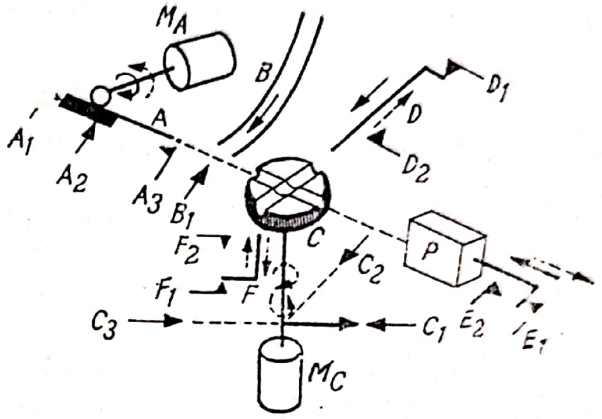


Fig. 11.73.— Schema cinematică pentru comanda automată a unei mașini de găurit.

b) Tija împingătoare A înaintează pînă în poziția extremă A_3, pentru care piesa este în poziția de găurire în canalul platanului C, orientat în direcția C_1. În acest moment, o altă tijă împingătoare D părăsește poziția D_1 și apasă lateral piesa care se găsește în canalul platanului. Simultan, perforatorul P părăsește poziția E_1 și înaintînd către poziția E_2, găurește piesa.

c) După găurirea piesei, perforatorul P revine în poziția limită E_1, apoi tija împingătoare D revine în poziția D_1 și tija împingătoare A revine în poziția intermediară A_2. Apariția semnalului A_2 determină rotirea platanului C cu o semitură spre dreapta astfel încît el se orientează către opritorul C_3, pregătind astfel pentru găurire a doua față a piesei.

d) Tijele împingătoare A și D revin în pozițiile A_3 și D_1 iar perforatorul P efectuează cea de a doua găurire, apoi cele două tije împingătoare se retrag în pozițiile limită simultan cu deplasarea unui opritor din F_1 către F_2.

e) Programul de lucru nu se modifică dacă limitatorul F_1 este acționat înainte ca $A_2 = 0$, sau dacă limitatorul F_2 este acționat înainte ca $A_2 = 0$.

f) Imediat ce apare semnalul A_2 și D_1, platanul C începe să se rotească spre stînga cu un sfert de tură și vine în direcția C_2, poziția lui fiind precizată prin opritorul F. Urmează deplasarea

tijei împingătoare din D_1 în D_2, care astfel ejectează piesa găurită, reîntoarcerea tijei D în poziţia iniţială, rotirea platanului C spre stînga cu un sfert de tură, pînă în poziţia iniţială C_1 şi apoi reîncepe un nou ciclu de lucru.

11.5.1. CODIFICAREA SPECIFICAŢIILOR SCHEMEI

Analizînd schema cinematică a automatului, dată în figura 11.73, putem stabili de la început următorul raport de stare generic :

$$\left(\frac{xA_1 \cdot A_2A_3B_1 \cdot C_1C_2C_3 \cdot D_1D_2E_1 \cdot E_2F_1F_2}{XA_s \cdot A_dDE \cdot C_sC_dF}\right)_n .$$

Variabila secundară x, ataşată variabilei de ieşire X, a fost de la început introdusă avîndu-se în vedere necesitatea memorării comenzii date prin B_1. Dacă este necesar, ulterior se pot adăuga şi alte variabile secundare.

S-a presupus că pentru comanda reversibilă a servomotoarelor M_A şi M_C se folosesc contactoarele A_s, A_d, respectiv C_s şi C_d, că pentru acţionarea tijelor împingătoare D şi F se folosesc servomotoare pneumatice, comandate prin amplificatoare bistabile cu ajutorul ventilelor electromagnetice D şi F şi că deplasarea perforatorului P se face cu ajutorul unui servomotor comandat de un amplificator monostabil prin intermediul ventilului electromagnetic E.

Codificat, specificaţiile a), b), c), şi d), se scriu astfel

$$\left(\frac{14.452}{0}\right)_1 \xrightarrow{B_1=1} \left(\frac{15.452}{200}\right)_2 \xrightarrow{x=1} \left(\frac{35.452}{240}\right)_3 \xrightarrow{B_1=0} \left(\frac{34.452}{240}\right)_4 \xrightarrow{A_1=0}$$

$$\left(\frac{25.452}{240}\right)_5 \xrightarrow{B_1=0} \left(\frac{24.452}{240}\right)_6 \xrightarrow{A_2=0} \left(\frac{20.452}{240}\right)_7 \xrightarrow{A_3=1} \left(\frac{22.452}{220}\right)_8 \xrightarrow{D_1=0}$$

$$\left(\frac{22.412}{230}\right)_9 \xrightarrow{E_1=0} \left(\frac{22.402}{230}\right)_{10} \xrightarrow{E_2=1} \left(\frac{22.406}{30}\right)_{11} \xrightarrow{x=0} \left(\frac{2.406}{20}\right)_{12} \xrightarrow{E_2=0}$$

$$\left(\frac{2.402}{20}\right)_{13} \xrightarrow{E_1=1} \left(\frac{2.412}{100}\right)_{14} \xrightarrow{D_1=1} \left(\frac{2.452}{100}\right)_{15} \xrightarrow{A_3=0} \left(\frac{452}{100}\right)_{16} \xrightarrow{A_2=1}$$

Scanned with OKEN Scanner

$$\left(\frac{4.452}{2}\right)_{17} \xrightarrow{C_1=0} \left(\frac{4.052}{2}\right)_{18} \xrightarrow{C_2=1} \left(\frac{4.252}{2}\right)_{19} \xrightarrow{C_2=0} 4.052 \xrightarrow{C_3=1}$$

$$\left(\frac{4.152}{40}\right)_{20} \xrightarrow{A_2=0} \left(\frac{152}{41}\right)_{21} \xrightarrow{A_3=1} \left(\frac{2.152}{21}\right)_{22} \xrightarrow{D_1=0} \left(\frac{2.112}{31}\right)_{23} \xrightarrow{E_1=0}$$

$$\left(\frac{2.102}{31}\right)_{24} \xrightarrow{E_2=1} \left(\frac{2.106}{231}\right)_{25} \xrightarrow{x=1} \left(\frac{22.106}{221}\right)_{26} \xrightarrow{E_2=0} \left(\frac{22.102}{221}\right)_{27} \xrightarrow{E_1=1}$$

$$\left(\frac{22.112}{301}\right)_{28} \xrightarrow{D_1=1} \left(\frac{22.152}{301}\right)_{29} \xrightarrow{A_3=0} \left(\frac{20.152}{301}\right)_{30} \xrightarrow{A_2=1} \left(\frac{24.152}{305}\right)_{31} \xrightarrow{A_1=1}$$

$$\left(\frac{34.152}{205}\right)_{32} \ldots\ldots$$

Specificația e), se scrie codificat astfel

$$4.152 \xrightarrow{F_1=0} \left(\frac{4.150}{41}\right)_{33} \xrightarrow{A_2=0} \left(\frac{150}{41}\right)_{34} \xrightarrow{A_3=1} \left(\frac{2.150}{21}\right)_{35} \xrightarrow{D_1=0}$$

$$\left(\frac{2.110}{31}\right)_{36} \xrightarrow{E_1=0} \left(\frac{2.100}{31}\right)_{37} \xrightarrow{E_2=1} \left(\frac{2.104}{231}\right)_{38} \xrightarrow{x=1} \left(\frac{22.104}{221}\right)_{39} \xrightarrow{E_2=0}$$

$$\left(\frac{22.100}{221}\right)_{40} \xrightarrow{E_1=1} \left(\frac{22.110}{301}\right)_{41} \xrightarrow{D_1=1} \left(\frac{22.150}{301}\right)_{42} \xrightarrow{A_3=0} \left(\frac{20.150}{301}\right)_{43} \xrightarrow{A_2=1}$$

$$\left(\frac{24.150}{305}\right)_{44} \xrightarrow{A_1=1} \left(\frac{34.150}{205}\right)_{45} \ldots\ldots$$

sau

$$4.150 \xrightarrow{F_2=1} \left(\frac{4.151}{41}\right)_{46} \xrightarrow{A_2=0} \left(\frac{151}{41}\right)_{47} \xrightarrow{A_3=1} \left(\frac{2.151}{21}\right)_{48} \xrightarrow{D_1=0}$$

$$\left(\frac{2.111}{31}\right)_{49} \xrightarrow{E_1=0} \left(\frac{2.101}{31}\right)_{50} \xrightarrow{E_2=1} \left(\frac{2.105}{231}\right)_{51} \xrightarrow{x=1} \left(\frac{22.105}{221}\right)_{52} \xrightarrow{E_2=0}$$

$$\left(\frac{22.101}{221}\right)_{53} \xrightarrow{E_1=1} \left(\frac{22.111}{301}\right)_{54} \xrightarrow{D_1=1} \left(\frac{22.151}{301}\right)_{55} \xrightarrow{A_3=0} \left(\frac{20.151}{301}\right)_{56} \xrightarrow{A_2=1}$$

$$\left(\frac{24.151}{305}\right)_{57} \xrightarrow{A_1=1} \left(\frac{34.151}{205}\right)_{58} \ldots\ldots$$

Scanned with OKEN Scanner

În fine, specificaţia f), se scrie codificat astfel

$$24.151 \xrightarrow{C_3=0} \left(\frac{24.051}{305}\right)_{59} \xrightarrow{A_1=1} \left(\frac{34.051}{205}\right)_{60} \ldots\ldots$$

sau

$$24.051 \xrightarrow{C_2=1} \left(\frac{24.251}{305}\right)_{61} \xrightarrow{A_1=1} \left(\frac{34.251}{221}\right)_{62} \ldots\ldots$$

şi

$$34.251 \xrightarrow{D_1=0} \left(\frac{34.211}{221}\right)_{63} \xrightarrow{D_2=1} \left(\frac{34.231}{21}\right)_{64} \xrightarrow{x=0} \left(\frac{14.231}{0}\right)_{65} \xrightarrow{D_2=0}$$

$$\left(\frac{14.211}{0}\right)_{66} \xrightarrow{D_1=1} \left(\frac{14.251}{0}\right)_{67} \xrightarrow{F_2=0} \left(\frac{14.250}{0}\right)_{68} \xrightarrow{F_1=1} \left(\frac{14.250}{4}\right)_{69} \xrightarrow{C_2=0}$$

$$\left(\frac{14.052}{4}\right)_{70} \xrightarrow{C_1=1} 14.452 \ldots\ldots$$

11.5.2. STRUCTURAREA SIMBOLURILOR DE MARCARE

Avînd în vedere observaţiile făcute la exemplul anterior de proiectare, vom exemplifica numai procedeul folosit pentru determinarea simbolului de marcare al variabilei de ieşire X, reprezentată în numitorul raportului de stare prin cifra de pondere 200.

Inspectînd stările secvenţei de lucru a automatului constatăm că variabila X îşi modifică valoarea în cazul tranziţiilor între stările 1—2, 10—11, 24—25, 37—38, 50—51 şi 63—64, tranziţii determinate de intervenţia variabilelor B_1, D_2 şi E_2. Adăugînd variabila secundară x, obţinem şirul $xB_1D_2E_2$. Ponderile acestor variabile sînt cuprinse în numărul octal 21.024. Folosind numai cifrele acestui număr, stările afectate de schimbarea valorilor binare ale variabilelor din şirul scris mai sus, se vor scrie

$X = 0 : (0)_1, (20.004)_{11}, (4)_{12}, (0)_{13}, (20.020)_{64}, (20)_{65}, (0)_{66}.$

$X = 1 : (1.000)_2, (21.000)_3, (20.000)_4, (20.000)_6, (4)_{25}, (20.004)_{26}, (20.000)_{27}, (4)_{38}, (20.004)_{39}, (20.000)_{40}, (4)_{51}, (20.004)_{52}, (20.000)_{53}.$

Observăm că este necesar să discriminăm starea $(22.406)_{11}$ de stările $(22.106)_{26}$, $(22.104)_{39}$, și $(22.105)_{52}$ și starea $(2.406)_{12}$ de stările $(2.106)_{25}$, $(2.104)_{38}$ și $(2.105)_{51}$.

Adăugarea variabilei C_3, avînd ponderea 100, rezolvă problema. Într-adevăr, în acest caz, vom scrie

$$(20.104)_{26},\ (20.104)_{39},\ (20.104)_{52},\ (104)_{25},\ (104)_{38},\ (104)_{51}.$$

Adăugarea variabilei C_3, impune și adăugarea stărilor afectate de schimbarea acestei variabile, la șirul stărilor care definesc pe X. Acestea sînt

$$X = 0 : (100)_{20} \text{ și } X = 1 : (20.000)_{59}, (20.100)_{27}, (20.100)_{40}, (20.100)_{53}.$$

Indicele superior al simbolului de marcare, va fi deci:

$$xB_1 \cdot C_3D_2E_2.$$

Numerele de stare cuprinse de indicele inferior al simbolului de marcare se obține luînd în considerare toate stările care definesc pe X și efectuînd transformările de ponderi

$$20.000 \rightarrow 20,1.000 \rightarrow 10,100 \rightarrow 4,20 \rightarrow 2 \text{ și } : 4 \rightarrow 1.$$

Rezultă astfel

$$X = D_{1 : (5, 10, 20, 24, 25, 30)\ 0 : (0, 1, 2, 4, 21, 22).}^{xB_1 \cdot C_3D_2E_2}$$

Folosind același procedeu, obținem

$$A_s = D_{1 : (5, 43, 51, 53)\ 0 : (2, 3, 4, 11, 15, 13, 31, 35, 42, 44, 45, 55, 71, 73, 75).}^{xA_1A_2 \cdot C_1C_3E_1}$$

$$A_d = D_{1 : (1, 12)\ 0 : (0, 2, 5, 6, 10, 11, 15, 16)}^{x \cdot A_3C_1C_3}$$

Scanned with OKEN Scanner

$$D = D^{xA_1A_3 \cdot C_1C_2E_1}_{1:(10,\,11,\,14,\,50,\,54,\,55,\,63)\;0:(1,\,3,\,5,\,15,\,21,\,23,\,25,\,41,\,43,\,45,\,51,\,61,\,65)}.$$

$$E = D^{x \cdot C_1C_3D_1}_{1:(2,\,14)\;0:(0,\,1,\,3,\,4,\,5,\,10,\,11,\,12,\,13,\,15)}.$$

$$C_s = D^{xA_1A_2 \cdot C_1C_2F_1}_{1:(31,\,33,\,50,\,51,\,52,\,70,\,71)\;0:(0,\,2,\,5,\,10,\,11,\,13,\,15,\,32,\,35,\,40,\,41,\,45,\,55,\,72,\,75)}.$$

$$C_a = D^{x \cdot A_1A_2C_3}_{1:(2)\;0:(0,\,1,\,3,\,6,\,10,\,11,\,12,\,13,\,16,\,17)}.$$

$$F = D^{xA_2C_1 \cdot C_3F_1F_2}_{1:(4,\,5,\,6,\,24,\,25,\,44,\,45,\,46,\,61,\,65,\,66)\;0:(12,\,20,\,21,\,22,\,26,\,32,\,52,\,72)}.$$

11.5.3. DEDUCEREA EXPRESIILOR LOGICE MINIMALE

Vom exemplifica procedeul folosit numai pentru cazul variabilei X. Pentru această variabilă se formează grupările:

2 10	1 2	4 10
20 5/25 / 7 / 15 17	4 10/14 / 11 / 12 13	20 / 24 / 30 34
. . . . .	20 30	
	34	
$\underline{00}.1\underline{01} \rightarrow C_3E_2$	$\underline{0}1.\underline{000} \rightarrow B_1$	$1\underline{0}.\underline{0}00 \rightarrow x\overline{D}_2\overline{E}_2$.

Schema corespunzătoare numai acestor grupări va avea hazard de comutaţie întrucît prima şi ultima grupare nu are nici-un număr de stare comun. Pentru eliminarea acestuia se impune deci adăugarea grupării

1 2	
10 24/34 / 25 / 26 27	$1\underline{0}.1\underline{00} \rightarrow xC_3$.
. . . . .	

Rezultă deci soluția minimală

$$X = C_3E_2 \cup B_1 \cup x\overline{D}_2\overline{E}_2 \cup (xC_3) = C_3E_2 \cup B_1 \cup x(C_3 \cup \overline{D}_2\overline{E}_2).$$

Folosind același procedeu, obținem

$$A_s = \bar{x}\overline{A}_2C_1E_1 \cup x\overline{A}_1\overline{C}_1E_1 \cup (A_1\overline{A}_2) \cup (\overline{A}_2\overline{C}_1\overline{C}_3) =$$

$$= E_1(\bar{x}\overline{A}_2C_1 \cup x\overline{C}_1\overline{A}_1) \cup \overline{A}_2(A_1 \cup \overline{C}_1\overline{C}_3).$$

$$A_d = \bar{x}\overline{A}_3C_3 \cup x\overline{A}_3C_1 \cup (C_1C_3) = \overline{A}_3(\bar{x}C_3 \cup xC_1) \cup C_1C_3.$$

$$D = A_3(\bar{x}\overline{C}_1 \cup xC_1E_1) \cup \overline{E}_1 \cup xA_1C_2.$$

$$E = \bar{x}C_3\overline{D}_1 \cup xC_1\overline{D}_1 \cup (C_1C_3) = D_1(\bar{x}C_3 \cup xC_1) \cup C_1C_3.$$

$$C_s = A_1\overline{C}_1F_1 \cup x\overline{A}_1A_2\overline{C}_1 \cup A_2\overline{C}_1\overline{C}_2 \cup (xA_2\overline{C}_1\overline{C}_2) =$$

$$= xA_2\overline{C}_1(\overline{C}_2 \cup \overline{A}_1) \cup \overline{C}_1(A_1F_1 \cup A_2\overline{C}_2).$$

$$C_d = x\overline{A}_1A_2\overline{C}_3.$$

$$F = \overline{C}_1(\overline{A}_2 \cup x) \cup C_3\overline{F}_1.$$

11.5.4. STRUCTURAREA SCHEMEI DE COMUTAȚIE

Pe baza formulelor deduse se poate structura schema de comutație dinamică sau de comutație statică care satisface specificațiile automatului.

Scanned with OKEN Scanner

În figura 11.74, este dată schema de comutaţie dinamică. Observăm că în schemă sînt folosite mai multe contacte ale aceluiaşi microîntrerupător. Aceasta impune folosirea unor relee intermediare pentru microîntrerupătoarele A_1, A_2, A_3, C_1, C_2, C_3, şi E_1. Se are în vedere faptul că microîntrerupătoarele E_2 şi F_1 sînt executate sub forma asociaţiei dintre un contact normal închis şi unul normal deschis.

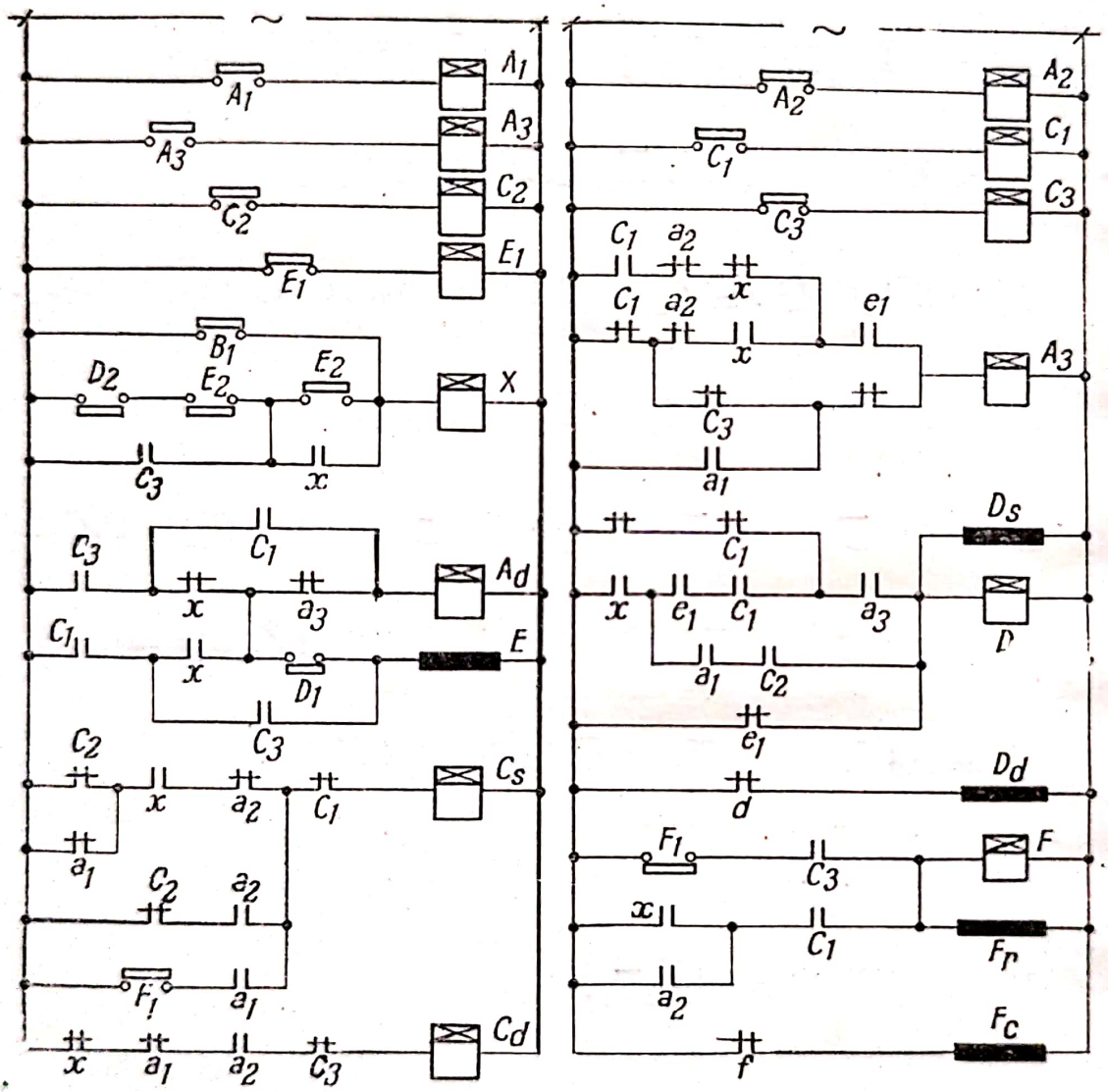


Fig. 11.74. — Comanda automată a unei maşini de găurit.

Odată structurată schema π, se impune verificarea funcţionării ei. Cu această ocazie se poate depista prezenţa hazardului care eventual a fost scăpat din vedere la formarea grupărilor de numere de stare adiacente.

Unele contacte sau grupuri de contacte pot fi folosite în comun de mai multe variabile de ieșire, dacă noile ramuri introduse prin aceasta, nu influențează valorile binare ale variabilelor de ieșire.

Astfel, în cazul variabilei X, contactul C_3 poate fi folosit în comun de două ramuri, prin aceasta expresia logică care definește pe X nemodificîndu-se

$$X = (E_2 \cup x)(\overline{D}_2\overline{E}_2 \cup C_3) \cup B_1 = \overline{D}_2\overline{E}_2x \cup C_3E_2 \cup C_3x \cup B_1.$$

Din aceleași motive contactul C_1 poate fi folosit în comun de două ramuri ale circuitului lui A_s, contactul x, este folosit în comun de două ramuri ale lui D și contactul C_1, este folosit în comun de două ramuri ale lui C_s.

Grupul de contacte : $(\bar{x}C_3 \cup xC_1) = a$, este comun variabilelor A_d și E. Folosirea în comun de către cele două variabile de ieșire a acestui grup de contacte nu modifică expresiile logice ale acestora, întrucît

$$a\overline{A}_3 \cup C_1C_3 \cup \overline{D}_1C_1C_3 = a\overline{A}_3 \cup C_1C_3$$

și

$$a\overline{D}_1 \cup C_1C_3 \cup \overline{A}_3C_1C_3 = a\overline{D}_1 \cup C_1C_3.$$

Folosirea în comun a contactelor C_1 și C_3 de asemenea nu modifică valoarea logică a funcțiilor logice pentru A_d și E, întrucît

$$C_1C_3 \cup C_1\bar{x}xC_1 \cup C_1\bar{x}C_3C_1 = C_1C_3$$

și

$$C_1C_3 \cup C_3x\bar{x}C_3 \cup C_3xC_1C_3 = C_1C_3.$$

La structurarea schemei din figura 11.74, s-a mai presupus că ventilele electromagnetice ale amplificatoarelor pneumatice sînt acționate prin electromagneții D_s, D_d, respectiv F_d, F_s și electromagnetul E. Revenirea organului de execuție al servomotorului E, cînd $E = 0$, se face sub acțiunea unui resort.

Scanned with OKEN Scanner

BIBLIOGRAFIE

1. Aizerman M. A., *Loghika, avtomati, algoritmi,* Moskva, Fiz.-mat. izd.-vo, 1963.
2. Anesa, Castello, Lamouche, *L'analyse séquentielle des schémas d'éléments statiques,* Equipement Electrique et électronique, **271** (1967).
3. Argintaru I., Cristina L., *Elemente pentru scheme de prelucrare logică și elemente de intrare în sistemul UNILOG tip A.*, Aut. și electr., **X**, *5* (1966).
4. Balaban E., Boțan N. V., *Aspecte noi privind extinderea aplicării algebrei logicii la proiectarea schemelor de acționare electrică,* Bul. I.P.I., **XII (XVI)**, *1—2*, 275—282 (1966).
5. Balaban E., *Contribuții la aplicarea calculului algebric în analiza și sinteza schemelor de comandă a acționărilor electrice,* — teză de doctorat, Institutul politehnic Iași, 1968.
6. Balaban E., Irhașiu M., Boțan C., Bordea D., *Studiu privind îmbunătățirea schemei de comandă pentru acționarea unei mașini-unelte,* Electrot., *16*, 2 (1968).
7. Barbu E., *Sinteza circuitelor logice de necoincidență și aplicarea lor,* Aut. și electr., *6* (1964).
8. Bartee T. C., *Computer design of multiple output logical networks,* IRE Trans. on El. Comp., *1*, 21—30 (1961).
9. Basarab S., *O metodă tip de sinteză a automatelor finite fără memorie cu mai multe ieșiri, realizate cu elemente de comutație statică,* Aut. și Electr., *2* (1965).
10. Bondarel R. ș.a., *Commande optimale des processus,* tome 1, *Concepts fondamentaux de l'automatique,* Paris, Dunod, 1967.
11. Boțan N. V., *Cum se citesc schemele de acționare electrică,* București, Edit. tehnică, 1965.
12. Boțan N. V., Bejan I., Balaban R., *Curs de acționări electromecanice și automatizări,* București, Edit. didactică și pedagogică, 1962.
13. Boțan N. V., Balaban E., *Sur quelques possibilités d'utilisation de l'algèbre logique aux schémas de commande électrique,* Bul. I.P.I., **XI (XV)**, *3 — 4* (1965).
14. Brunin J., *Logique binaire et commutation,* Paris, Dunod, 1966.
15. Brunin J., *Simplification des fonctions logiques à grand nombre de variables,* Automatisme. **V** (1963).

Scanned with OKEN Scanner

16. Caldwell S. H., *Switching Circuits and Logical Design*, Wiley and Sons, 1958.
17. Castello P., *Clé des schémas électriques*, Paris, Dunod, 1965.
18. Challe J. H., *Utilisation des éléments logiques dans les automatismes à programme par commutation*, Automatisme, **VI** (1961).
19. Creangă I. şi colectiv, *Introducere în teoria numerelor*, Bucureşti, Edit. didactică şi pedagogică, 1965.
20. Constantinescu V., *O metodă tip de simplificare a schemelor de comutaţie statică*, în *Probleme de automat.*, IV, Bucureşti, Edit. Academiei.
21. Costake N. N., Lazăr I. S., *Metodă de sinteză a circuitelor logice fără memorie*, Aut. şi electr., *5* (1962).
22. Damsker D., *Codurile folosite pentru reprezentarea informaţiilor în sistemele tehnice digitale*, A.T.M., *5* (1965).
23. Damsker D., *Bazele teoretice ale automatelor secvenţiale*, A.T.M., *6*, 33—44 (1966).
24. Damsker D., *Matrice logice complete. Folosirea lor în minimizarea expresiilor booleene*, Aut. şi electr., **X**, *3*, 97—148 (1966).
25. Damsker D., *Principii şi mijloace noi de automatizare a acţionărilor electrice*, Bucureşti, Edit. Academiei, 1964.
26. Damsker D., *Metodă de sinteză a sistemelor secvenţiale programabilă pe calculatoarele electronice numerice*, Aut. şi electr., **9**, *5*, 207—211 (1965).
27. Davidoviciu A., *Utilizarea elementelor de comutaţie statică UNILOG tip A în instalaţiile de automatizare*, Aut. şi electr., **X**, *5*, 230—236 (1966).
28. Denouette M., Perrin J. P., Daclin E., *Aléas dans les systèmes électroniques en fonctionnement séquentiel asynchrone*, Automatisme, **XII**, *4*, 166—173 (1967).
29. Didenko V. P., *Ob odnom metode sinteza bezpovtornîh kontaktnîh shem*, Avtomaticeskoe upravlenie, Izv. ANSSSR, 1960.
30. Dobrovidov A. V., Stratonovici R. L., *O sinteze optimalnîh avtomatov, funcţioniruiuşcih v sluciainîh sredah*, Avt. i telemeh., **10** (1964).
31. Enescu Gh., *Introducere în logica matematică*, Bucureşti, Edit. ştiinţifică, 1965.
32. Felea I., Ionescu C., Tecuceanu M., *Circuite şi blocuri pentru funcţiuni dependente de timp*, Aut. şi electr., **X**, *5*, 218—225 (1966).
33. Felea I., *Circuite cu transistoare în industrie*, II, Ed. tehnică, Bucureşti, 1964.
34. Florine J., *Méthode d'optimisation simultanée des fonctions de sortie d'un système logique incomplètement défini*, Revue A., *1* (1965).
35. Florin J., *La synthèse des machines logiques*, Paris, Dunod, 1964.
36. Florine J., *Sintez loghiceskih ustroistv i ego avtomatizaţiia*, Moskva, Izd. «Mir», 1966.
37. Florine J., *Automatisation des processus d'analyse et de synthèse optimale des fonctions booléennes*, Automatisme, **XI** (1966).
38. Gavrilov M. A., *Teoriia releino-kontaktnîh shem*, Moskva, Izd. AN SSSR, 1950.

Scanned with OKEN Scanner

39. GILL A., *Introduction to the theory of finite state machines*, McGraw-Hill, 1962.
40. GIRARD P., *Contribution à l'étude des machines séquentielles*, Automatisme, XIV, *2*, 5—80 (1969).
41. GIRARD P., *Contribution à l'étude des machines séquentielles* (II). Automatisme, **XV**, *1*, 12—16 (1970).
42. GORIOVIȚKAIA N. A., PINSKER E. L., *Postroenie minimalnîh diziunctivnîh normalnîh form*, Avt. upravlenie, Izd. AN SSSR, 1960.
43. GUY BOULAYE, *Application des équations booléennes à l'étude des bascules asynchronisées*, Automatisme, **XIII**, *1*, 20—24 (1968).
44. GUY BOULAYE, *Homomorphismes de récurrences. Application à la décomposition des systèmes séquentiels*, Automatisme, **XII**, *3*, 106—109 (1967).
45. HARTMANIS J., *On the state assignment problem for sequantial machines*, IRE Trans. EC., **10**, 157—165 (1961).
46. HILL C. F., *Nor/Nand Logic the Easy Way*, Control Engineering, mai, 1964.
47. HUFFMAN S., *The synthesis of sequential switching circuits*, Journ. of the Franklin Inst., mart.-apr. 1954.
48. JOFFRIN Y., PERRIN J. P., DENOUETTE R., *Automatismes à séquences*, Automatisme, **X** (1965).
49. KÄMMERER W., *Zur Struktur selbstkorrigierender Automaten*, Messen-Steuern-Regeln, ianuarie 1967.
50. KAZAKOV V. D., *Nahojdenie maksimalnogo cisla prostîh implikantov proizvolnoi loghiceskoi funkții n-peremennîh*, Avt. upravlenie, Izd. AN SSSR, 1960.
51. KRIGMAN A., *Bond Graphs for Logic Circuits*, Control Enginering, februarie, 91.92 (1967).
52. KUNTZMANN J., *Un théorème sur les composants premiers d'une fonction booléenne et ses applications*, Automatisme, *1*, 18—20 (1964).
53. KUNTZMANN J., *Algèbre de Boole*, Paris, Dunod, 1965.
54. KUNTZMANN J., *Synthèse optimale de fonctions booléennes par des opérateurs croissants*. Automatisme, **X** (1965).
55. ITSUO TABANAMÎ, NAMIO HONDA, *Analysis and Synthesis of Finite Automata by Transition Matrices* (abstracts), The Transaction of the Institute for Electronics and Comunication Engineers of Japan, 1967.
56. LAZĂR I. S., *Sinteza schemelor logice fără memorie, realizate cu elemente de comutație statică*, A.T.M., *6*, București, Edit. tehnică, 1966.
57. LAZAROV V. G., PILL P. I., *Sintez asinhronnîh konecinîh avtomatov*, Moskva, Izd-Vo « Nauka », 1964.
58. LEBORGNE P., TREMEMBERT A., *Relee electromagnetice și elementele de comutație statică în circuitele de automatizare*, Mec. aut. electronică, caiet selectiv, *8*, 504—514 (1967).
59. LAPSCHER F., *Fonctions booléennes incomplètes rattachées à une fonction donnée. Applications aux problèmes de représentation*, Automatisme, **X** (1965).

Scanned with OKEN Scanner

60. Lazarev V. G., *Metodă matricolă pentru minimizarea stărilor interioare ale automatelor asincrone finite*, Izv. AN SSSR, Tehn. kibernetika, *1*, 36—38 (1964).
61. Livovschi L., *Circuite cu contacte de relee*, București, Edit. Academiei, 1968.
62. Livovschi L., *Utilizarea grefelor în probleme de sinteză a automatelor finite*, Avt. i telemeh., **26**, *5*, 832—843 (1965).
63. Marinescu N., Barbu E., *Teoria circuitelor de comutație*, București, Edit. didactică și pedagogică, 1964.
64. Mergler H. W., *A systematic approach. The key to optimum logic design*, Control Engineering, *1*, 68 (1967).
65. Moisil Gr., *Teoria algebrică a circuitelor cu tranzistoare*, vol. I și II, București, Edit. tehnică, 1961—1962.
66. Moisil Gr., *Teoria algebrică a mecanismelor automate*, — București, Edit. tehnică, 1959.
67. Moisil Gr., *Scheme de comandă directă cu contacte și relee*, București, Edit. Academiei, 1959.
68. Moisil Gr., *Funcționarea în mai mulți timpi a schemelor cu relee ideale*, București, Edit. Academiei, 1960.
69. Moisil Gr., *Sur la théorie algébrique des circuits logiques à transistors*, Automatisme, **VII** (1962).
70. Moisil Gr., *Încercări vechi și noi de logică neclasică*, București, Edit. științifică, 1965.
71. Moisil Gr., *Sur les aléas de continuité dans les circuits de commutation*, Automatisme, **X** (1965).
72. Naslin P., *Circuite logice și automatizări secvențiale*, București, Edit. tehnică, 1967.
73. Naslin P., *Note sur la simplification des fonctions logiques*, Automatisme, *11*, 433—439 (1962).
74. Naslin P., *Classification des circuits séquentiels*, Automatisme, **IX**, *6*, 239—248 (1964).
75. Naslin P., *Note sur la recherche des bases premières d'une fonction booléenne*, Automatisme, **XII**, *1*, 16 (1967).
76. Necula N. N., *Principalele probleme ale teoriei circuitelor logice cu prag*, Aut. și electr. **12**, *3*, 112—119 (1968).
77. Necula N. N., *O metodă de minimizare a structurilor în două nivele SI—SAU de scheme combinaționale cu mai multe ieșiri*, Aut. și electr., **13**, *1*, 17—23 (1969).
78. Nicolau Ed., Popovici Al., *Introducere în cibernetica sistemelor discrete*, București, Edit. tehnică, 1966.
79. Novikov P. S., *Elemente de logică matematică*, București, Edit. științifică, 1966.
80. Piili E. I., *Nekotorîe voprosî kodirovaniia konecinîh avtomatov*, Avt. upravlenie, Izd. AN SSSR, 1960.
81. Pomper V., *Proposition de définition et de symbolisation des opérations et fonctions logiques*, Automatisme, **XI**, *5*, 304—308 (1966).

Scanned with OKEN Scanner

82. Pospelov D. A., *Loghiceskie metodi analiza i sinteza shem*, Moskva, Izd. vo «Energhetika», 1964.
83. Perrin J. P., Denouette M., Daclin E., *Systèmes logiques*, vol. I, Paris, Dunod, 1967.
84. Perrin J. P., Denouette M., Daclin E., *Systèmes logiques*, vol. II, Paris, Dunod, 1967.
85. Predoi A., Dragoș D., *Elaborarea schemelor logice combinaționale optimale pentru echipamentele numerice ale mașinilor unelte și elementele necesare acestor scheme*, Aut. și electr., **12**, *3*, 107—111 (1968).
86. Reus N., *O nouă metodă grafo-simbolică de analiză și sinteză a sistemelor discrete de acționare electrică*, teză de doctorat, Instit. politehnic Iași, 1960.
87. Richalet J., *Les systèmes discrets*, L'Onde Electrique, **451**, 1011—1010 (1964).
88. Richalet J., *Note sur l'application de la théorie des systèmes linéaires séquentiels à la reconnaissance des suites booléennes*, Automatisme, **XI** (1966).
89. Sandu D. ,*Utilisation des graphes linéaires dans la synthèse logique structurale*, An. șt. Univ., Iași, *XIII*, 89—101 (1967).
90. Sandu D., *O nouă metodă de sinteză logică structurală a automatelor finite combinaționale cu mai multe ieșiri*, în *Probleme de aut.*, vol. VI, București, Edit. Academiei.
91. Săvescu M., Popovici Al., Popescu M., *Circuite electronic e* II, Ed. tehnică, București, 1969.
92. Seewaldt H., *Scheme electrice de acționare a mașinilor și agregatelor*, București, Edit. tehnică, 1966.
93. Stănciulescu Fl., *O metodă algebrică de sinteză a schemelor secvențiale cu relee fără contacte*, Aut. și electr., **VII**, *6*, 250—254 (1964).
94. Stănciulescu Fl., *La conception et la réalisation des schémas logiques à grande fiabilité*, Automatisme, **XI**, *11*, 597—602 (1966).
95. Tanasiciuc Cl., Sîmbotin C., *Comutația statică în automatică și telemecanică*, București, Edit. tehnică, 1965.
96. Tanasiciuc Cl., Stănciulescu Fl., *Contribuții la sinteza schemelor secvențiale binare*, Aut. și electr., **6**, *1*, 197—201 (1962).
97. Thelliez S., *Simulateur de circuits logiques ternaires*, Automatisme, **XI**, *5*, 296—300 (1966).
98. Tison P., *Recherche des termes premiers d'une fonction booléenne*, Automatisme, ianuarie 1964.
99. Tison P., *Circuits logiques sous aléa de fonctionnement*, Automatisme, **XI**, *4*, 231—236 (1966).
100. Tison P., *Algèbre booléenne. Théorie des consensus. Recherche des bases premières d'une fonction*, Automatisme **X** (1965).
101. Turuta E. N., *Nekotorie metodi povișeniia nadejnosti konecinih avtomatov*, Avt. upravlenie, Izd. AN SSSR, 1960.

Scanned with OKEN Scanner

102. Unger S. H., *Hazards and relays in asynchronous sequentiel switching circuits*, IRE Trans., CT. 6, martie 1959.
103. Valée R., *Analyse binaire symbolique*, Automatisme, **VII** (1962).
104. Vasilieva N. P., Taşkoveţ I. S., *Loghiceskie elementi v promişlennoi avtomatike*, Bibl. po avtomatike, 1962.
105. Vasilieva N. P., Taşkoveţ I. S., *Prosteişie posledovatelnostnie elementi i realizaţiia zapominaiuşcei ciasti posledovatelnostnih funkţii na etih elementah*, Avt. i telemeh., **6** (1964).
106. Zissos D., Copperhite G. V., *The design of minimal Nor/Nand logical circuits*, Electronic Engineering, septembrie 1965.
107. Zander H. J., Anack L., *Praktische Anwendung von Verfahren zur Synthese von Folgeschaltungen*, Messen-Steuern-Regeln, *8*, 370—376 (1965).
108. Zakrevski A. D., *O priblijennih metodah reşeniia, loghiceskih zadaci. Problemi sinteza ţifrovih avtomatov*, Moskva, Izd-vo « Nauka », 1967.

Scanned with OKEN Scanner

SYMBOLIC TREATMENT OF COMMUTATION SCHEMES

(SUMMARY)

The first chapters of this work define two symbolic forms of ciphred expression of the properties of a commutation scheme, namely the "marking symbol" and "state ratio".

The first symbolic form allows the establishment of certain correspondence relations useful in the process of analysis and synthesis of the commutation scheme, finally leading to the defining of a graph, specific of commutation schemes, and to a new criterion of classification of the logical functions.

The second symbolic form proves useful in describing the states of a commutation scheme, giving the possibility to deduce its progress with the time. The properties of the above mentioned graph allow the elaboration of a method for minimizing the Boolean expressions, which differ from both Quine and McCluskey methods through the mode of finding the prime implicants. This method proves very efficient, lends itself to engineering calculations and allows the solving of certain problems with more than six binary variables.

The work also considers the problem of minimizing the systems of Boolean functions and that of the transformation of the schemes of dynamic commutations.

The last chapters deal with the solving of the problem of the analysis and synthesis of the commutation schemes under an applicative aspect.

Thus, besides the solving of 9 simpler analysis problems and 31 simpler synthesis problems, taken from the field of automatic electric driving of machines with programmed kinematics and of computers, three other problems of greater complexity, pertaining to industry, are treated. Thus the analysis of Platts-type automatic card and the logical plan of the automation schemes for a cotton rolling mill and for a self-acting drill are made. In the last case, the

Scanned with OKEN Scanner

number of the input binary variables of the commutation scheme is 14. The examples treated may constitute a guide for solving certain similar problems.

The theoretical part of the work is accompanied by numerous exemplifications and in the applicative part certain particular cases are chosen, solutions being thus giren for various specific situations that may practically appear.

In order to represent the static commutation scheme, a representation under the form of an oriented graph has been adopted, this way of representing being the simplest possible.

The whole book has a unified character, trying to point out the utility of using the 2 symbolic forms of ciphred expression of the properties of the finite automata.

Scanned with OKEN Scanner

Redactor: Ing. LIA CREȚOIU
Tehnoredactor: MARILENA POPESCU

Bun de tipar 12.04.1971. Hîrtie scris I A. Format 16/61×86, de 80 G/m². Coli de tipar 17. C.Z. pentru biblioteci mari 51:621.316.5.06.001.2. C.Z. pentru biblioteci mici 51:621.316.

Întreprinderea Poligrafică „Informația", București,
str. Brezoianu nr. 23–25.
Republica Socialistă România

Scanned with OKEN Scanner

621.394

Lei 11

EDITURA
ACADEMIEI
REPUBLICII
SOCIALISTE
ROMÂNIA

Scanned with OKEN Scanner

În primele capitole ale acestei lucrări se definesc două forme simbolice de exprimare codificată a proprietăților unei scheme de comutație și anume **„simbolul de marcare"** și **„raportul de stare"**.

Prima formă simbolică permite stabilirea unor relații de corespondență utile în procesele de analiză și sinteză a schemelor de comutație, conducînd în final la definirea unui graf specific schemelor de comutație și la un nou criteriu de clasificare a funcțiilor logice.

A doua formă simbolică se dovedește utilă în descrierea stărilor unei scheme de comutație, oferind posibilitatea de a deduce evoluția ei în timp.

Proprietățile grafului specific schemelor de comutație permit elaborarea unei metodici de minimizare a expresiilor booleene care diferă de metoda lui Quine și McCluskey prin modul de găsire a implicanților primi. Această metodă se dovedește foarte expeditivă, se pretează calculului ingineresc manual și permite rezolvarea unor probleme cu mai mult de șase variabile binare.

În lucrare este abordată și problema minimizării sistemelor de funcții booleene precum și cea a transformării schemelor de comutație dinamică.
Ultimele capitole ale lucrării tratează problema analizei și sintezei schemelor de comutație sub un aspect inginerese aplicativ.